Noel M. Morris

ELEMENTI DI ELETTRONICA DIGITALE

Dispositivi semiconduttori e circuiti integrati · Porte logiche di base · I circuiti logici Porte logiche NAND e NOR · Elementi bistabili · La numerazione e l'aritmetica binaria · Circuiti contatori · Registri di scorrimento (shift register) Sistemi di memoria · Visualizzatori alfanumerici · Calcolatori e microprocessori · Attrezzature di collaudo

HOEPLI

ENCICLOPEDIA HOEPLI

in 14 volumi **completa e aggiornata**

In-4 (22×28 cm) di pagine LXIV-7028 contenenti 72910 voci, 88140 accezioni, 10340 disegni originali al tratto, 1202 riproduzioni di capolavori d'arte in 145 tavole fuori testo in rotocalcografia e 856 soggetti e disegni a colori in 76 tavole fuori testo. L'opera completa e aggiornata (marzo 1979), rilegata in tutto skivertex, titoli e fregi in oro fino

ISBN 88-203-1124-0 L. 400.000

MORRIS N.M., **Elementi di elettronica teorica e pratica.** Traduzione di *P. Adorni.* In-8, di pagine XIV-302, con 199 figure. Copertina a colori plasticata e telata **ISBN 88-203-1092-9 L. 8000**

Circuiti a corrente continua - Resistori - Condensatori - Induttori - Tensioni e correnti alternate - Circuiti a corrente alternata sinusoidale - Trasformatori - Diodi a stato solido - Transistori - Optoelettronica - Amplificatori e circuiti logici fondamentali - Microelettronica e circuiti integrati - Amplificatori a retroazione e oscillatori - Amplificatori operazionali - Alimentatori stabilizzati e elettronica di potenza - Apparecchi di misura.

BARANZINI R. e G. DUGNANI, **Microprocessori e microcomputers.** 2ª edizione riveduta e aggiornata. In-8, di pagine XIV-450, con 220 figure e 13 tabelle. Copertina a colori plasticata **ISBN 88-203-1198-4 L. 22000**

CERRUTTI R. e M. MOROCUTTO, **Introduzione ai microprocessori.** Il codice binario - I circuiti logici - Il microcomputer - La programmazione - I sistemi di sviluppo - Un esempio reale - Microprocessori in commercio -Glossario. In-8, di pagine VIII-112, con 33 figure e tabelle. Copertina a colori plasticata e telata **ISBN 88-203-1172-0 L. 6000**

COSTA E., **Radioriparatore:**

Volume I: "Audio radio". 10ª edizione completamente rifatta. In-8, di pagine XII-368, con 486 figure. Copertina a colori plasticata e telata **ISBN 88-203-0961-0 L. 12000**

Volume II: "Componenti-Misure-Allineamenti-Guasti". 10ª edizione completamente rifatta. In-8, di pagine XII-268, con 323 figure. Copertina a colori plasticata. **ISBN 88-203-1167-4 L. 15000**

GHERSEL F., **I cinescopi a colori**. Cinescopi ad un fascio - Cinescopi a tre fasci a delta - Cinescopi a tre fasci allineati - Fosfori - Funzionamento -Fabbricazione - Circuiti associati - Difetti - Misure - Definizioni - Nomenclatura in italiano, inglese, francese, tedesco. In-8, di pagine X-182, con 126 figure in bianco e nero e a colori e 7 tabelle. Copertina a colori plasticata e telata **ISBN 88-203-1160-7 L. 15000**

Schemario TVC. Servizio, misure e schemi di televisori a colori. In-4, di pagine X-214, con 150 figure in bianco e nero e a colori. Copertina a colori plasticata **ISBN 88-202-1213-1 L. 22000**

GRIFFIN G. e P. SAPORITO, **Videotape.** Guida all'uso dei mezzi televisivi leggeri. IN-8, di pagine VIII-216, con 97 illustrazioni di *Gabriel Griffin*. Copertina a colori plasticata **ISBN 88-203-1187-9 L. 12000**

MENDOLIA I., **Elettronica generale**, ad uso degli Istituti tecnici industriali. In-8, di pagine XVI-832, con 862 figure e 5 tavole. Copertina a colori plasticata **ISBN 88-203-1171-2 L. 20000**

MIOTTO M.,**Il televisore a colori PAL.** Circuiti a transistori e integrati -Sovraimpressioni - Giochi al televisore. In-8, di pagine XII-148, con 161 figure, 4 tabelle. Copertina a colori plasticata e telata **ISBN 88-202-1136-4 L. 10000**

POTENZA F., **La matematica per il calcolatore tascabile.** Problemi di lavoro, grafici, giochi, impostati e risolti col minicalcolatore. In-16, di pagine X-294, con 27 figure. Copertina a colori plasticata e telata **ISBN 88-203-1059-7 L. 4500**

RAVALICO D.E., **Il radiolibro.** Radiotecnica pratica. 19ª edizione ampliata e aggiornata a cura di Giorgio Terenzi. In-8, di pagine VIII-500, con 108 figure, 628 zoccoli e valvole di transistore di I.C., 65 tabelle e 5 tavole fuori testo. Copertina a colori plasticata e telata **ISBN 88-203-1044-9 L. 10000**

EDITORE ULRICO HOEPLI MILANO

ELEMENTI DI
ELETTRONICA DIGITALE

NOEL M. MORRIS

ELEMENTI DI ELETTRONICA DIGITALE

Dispositivi semiconduttori e circuiti integrati - Porte logiche di base
I circuiti logici - Porte logiche NAND e NOR
Elementi bistabili - La numerazione e l'aritmetica binaria
Circuiti contatori - Registri di scorrimento (shift register)
Sistemi di memoria - Visualizzatori alfanumerici
Calcolatori e microprocessori - Attrezzature di collaudo

EDITORE ULRICO HOEPLI MILANO

Indice generale

Capitolo 1

Dispositivi semiconduttori e circuiti integrati

Capitolo 2

Porte logiche di base

Capitolo 3
I circuiti logici

Capitolo 4
Porte logiche NAND e NOR

Capitolo 5

Elementi bistabili

Capitolo 6

Numerazione e aritmetica binaria

Capitolo 7

Circuiti contatori

Capitolo 8
Registri di scorrimento (shift register)

Capitolo 9
Sistemi di memoria

Capitolo 10
Visualizzatori alfanumerici

Capitolo 11

Calcolatori e microprocessori

Capitolo 12

Attrezzature di collaudo

Capitolo 1

Dispositivi semiconduttori e circuiti integrati

1.1 Elettroni e buche

Una corrente elettrica è spiegabile in termini di movimento di *cariche elettriche* tra due punti distinti di un circuito. Per spiegare il flusso di corrente è necessario sapere qualcosa sulla struttura atomica dei materiali usati nei circuiti elettronici.

Da un punto di vista ingegneristico, un atomo consiste di due tipi di particelle cariche, **elettroni** e **protoni**. La massa del protone è 1840 volte quella dell'elettrone; i protoni sono concentrati al centro dell'atomo, detto *nucleo*, come in Fig. 1.1. Un protone porta una carica elettrica po-
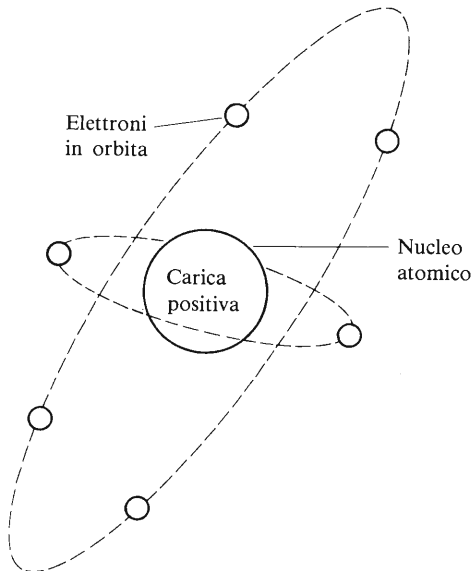
Fig. 1.1 Elettroni in orbita attorno al nucleo

sitiva, mentre l'elettrone ne porta una uguale ed opposta. Ogni atomo ha tanti elettroni quanti protoni, cosicché la carica elettrica complessiva è nulla. Se un atomo perdesse un elettrone, resterebbe con una carica positiva in eccesso.

Gli elettroni orbitano attorno al nucleo (Fig. 1.1) in quelle che sono definite **bande, strati** o **gusci**. Si può tracciare una semplice analogia tra gli strati degli elettroni e i livelli di parcheggio per auto in un posteggio a molti piani. Il piano terra può essere considerato come il nucleo dell'atomo, mentre i piani superiori rappresentano i livelli di energia in cui si possono trovare gli elettroni. In natura, i livelli d'energia più bassi dell'atomo devono essere pieni di elettroni prima che altri elettroni possano comparire nei livelli a contenuto d'energia superiore (nel caso del parcheggio, sono i piani inferiori ad essere riempiti per primi).

Quando un livello energetico è completo, si raggiunge una configurazione elettronica di grande stabilità, per cui non sono possibili, senza un elevato dispendio di energia, movimenti di elettroni al di fuori dell'orbita atomica corrispondente a ciascuno di essi. Tuttavia è probabile che i livelli energetici più elevati in cui si trovano gli elettroni, non siano completamente pieni. Il guscio più esterno dell'atomo che contiene almeno un elettrone è detto **banda di valenza** o **livello energetico di valenza**; gli elettroni che in definitiva prendono parte al processo di conduzione orbitano nella banda di valenza.

Perché un elettrone possa prendere parte alla conduzione, deve acquistare una quantità di energia superiore a quella necessaria a mantenerlo orbitante nella banda di valenza. L'energia può venire fornita da molte sorgenti, compresi il calore, la luce e le radiazioni atomiche. L'applicazione di una tensione elettrica a un semiconduttore dà come risultato l'applicazione di una forza elettrica agli elettroni della banda di conduzione, forza che li si spinge verso il polo positivo del generatore. Gli elettroni che arrivano al polo positivo costituiscono il flusso di corrente nel circuito, contemporaneamente un numero uguale di elettroni viene immesso nel circuito da parte del polo negativo della batteria per mantenere bilanciata la carica elettronica all'interno del semiconduttore.

Quando un elettrone sfugge a un atomo, diventa una carica elettrica negativa *mobile*; dato che l'atomo ha perso un elettrone (una carica negativa, cioè), rimane con una carica positiva netta corrispondente a un'unità di carica elettrica. Come risultato di questo, cercherà di attrarre a sé un elettrone mobile che si trovi nelle sue vicinanze. Perciò è possibile descrivere la carica positiva sull'atomo come una **buca** elettronica; una buca può essere considerata un portatore di carica positiva, così co-

me un elettrone è un portatore di carica negativa. *Una buca è semplicemente l'assenza di un elettrone da un punto della struttura dove normalmente se ne dovrebbe trovare uno.*

1.2 Semiconduttori

Un semiconduttore è un materiale la cui resistività è compresa tra quella di un buon conduttore e quella di un buon isolante. I materiali semiconduttori comunemente usati nei diodi e nei transistori sono il **silicio** e il **germanio**, mentre altri materiali come l'arseniuro di gallio e il fosfuro di gallio sono utilizzati in altri dispositivi come i diodi fotoemettitori (LED). Il silicio è l'elemento semiconduttore di gran lunga più usato e si trova in molti tipi di rocce e pietre (ad esempio, la sabbia è biossido di silicio).

Il passaggio di corrente in un semiconduttore puro è il risultato di movimenti di elettroni e buche generati principalmente dagli effetti termici come delineato nel paragrafo 1.1. All'aumentare della temperatura esterna, aumenta il numero di elettroni che possono prendere parte al fenomeno di conduzione. Quindi, per un determinato valore della tensione applicata, l'intensità di corrente attraverso un materiale semiconduttore aumenta con la temperatura, cioè la sua resistenza diminuisce con l'aumentare della temperatura. In altre parole, la resistività nei semiconduttori presenta un **coefficiente termico negativo**. Il flusso di corrente creato da questo effetto è, nel caso dei diodi e dei transistori, un inconveniente poiché dà origine a una corrente di perdita che dipende dalla temperatura. I dispositivi al germanio hanno un valore della corrente di perdita dovuta a questo effetto superiore rispetto a quelli al silicio. Questo tipo di conduttività è detto **conduttività intrinseca**; e per questa ragione, i semiconduttori puri sono detti semiconduttori di **tipo i**.

Per ottenere un valore di conduttività che possa essere controllato dall'esterno, si introducono nel semiconduttore, durante il processo di fabbricazione, quantità definite di impurità. La quantità tipica introdotta è di 1 parte ogni 10 milioni di parti di materiale puro. Secondo il tipo di impurità aggiunta il semiconduttore è detto tipo n o di tipo p.

In un materiale di **tipo p**, le impurità aggiunte provocano un eccesso di buche mobili. La dizione "di tipo p" implica che ci siano portatori di cariche **p**ositive. Il gallio e l'iridio sono materiali spesso usati nel drogaggio del silicio per produrre un semiconduttore di tipo p. Ricordiamo che in un pezzo isolato di materiale di tipo p ci sono un egual numero di elettroni e di buche, cosicché, considerandolo nel suo insieme, esso è

elettricamente neutro; tuttavia, non tutti i portatori di carica sono "liberi" di dar luogo a un flusso di corrente, dato che la maggior parte di essi è legata agli atomi da forze elettrostatiche di legame atomico. Così, la corrente in un materiale di tipo p è in gran parte dovuta al movimento di portatori mobili di cariche positive verso il polo negativo del generatore. Perciò nei semiconduttori di tipo p, i buchi sono chiamati **portatori di maggioranza**. Contemporaneamente per effetto termico viene generato un numero assai esiguo di elettroni mobili i quali danno un piccolo contributo all'intensità della corrente. In un materiale di tipo p gli elettroni sono perciò detti **portatori di minoranza**.

In un materiale di **tipo n**, la droga usata (solitamente arsenico o antimonio) provoca un eccesso di elettroni mobili (la dizione "di tipo n" implica che ci siano portatori di carica **n**egativa). Nei materiali di tipo n, la maggior parte dell'intensità di corrente è dovuta a movimenti di elettroni attraverso il materiale, e in *questo caso gli elettroni sono chiamati portatori di maggioranza.*

I movimenti delle buche (che sono i *portatori di minoranza in materiali di tipo n*), generate da effetti termici, danno anch'essi un piccolo contributo all'intensità di corrente.

Entrambi i semiconduttori di tipo p e di tipo n sono utilizzati nella fabbricazione dei dispositivi semiconduttori. Alcuni dispositivi particolari usano invece semiconduttori di tipo i.

1.3 Caratteristiche dei diodi

Un diodo è un dispositivo elettronico con due terminali (elettrodi) uno dei quali è detto **anodo** e l'altro **catodo** (Fig. 1.2a). Il diodo presenta una bassa resistenza al passaggio di corrente quando il potenziale anodico è positivo rispetto a quello catodico, e in questo caso si dice che è **polarizzato direttamente**. Quando, invece, il potenziale anodico è negativo rispetto a quello catodico, il diodo presenta una resistenza molto elevata al passaggio di corrente, e in questo caso si dice che esso è **polarizzato inversamente**. In tale senso il diodo può essere considerato come un **interruttore sensibile alla tensione**, che è chiuso quando è polarizzato direttamente e aperto quando è polarizzato inversamente.

In un diodo semiconduttore (vedi paragrafo 1.4) il catodo è generalmente contrassegnato con un punto o con una striscia colorata. I diodi possono essere provati usando un tester in funzione di ohmmetro. Bisogna notare che il terminale "+" di uno strumento del genere è in realtà negativo rispetto al terminale "−" per il fatto di essere collegato
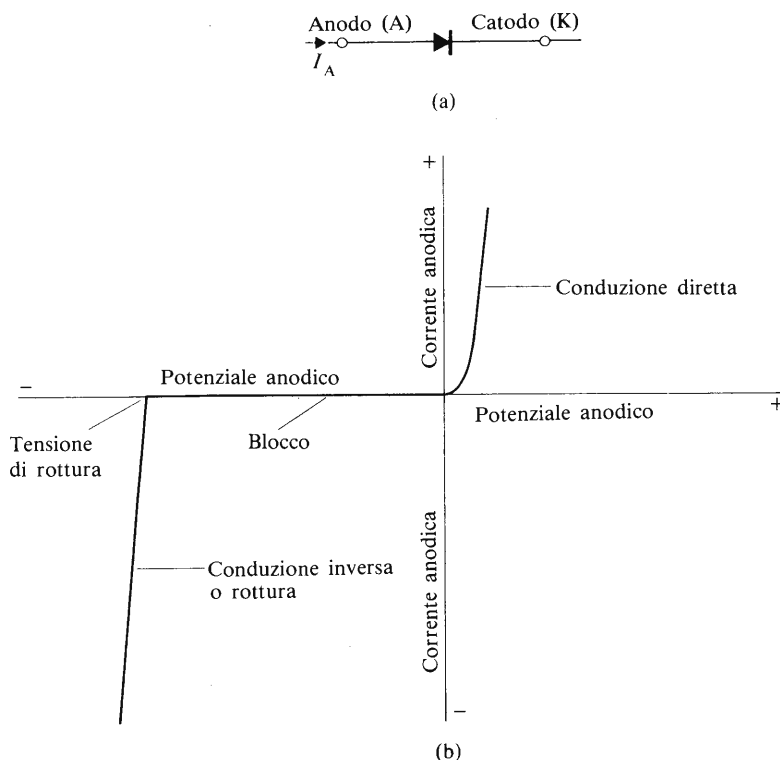
Fig. 1.2 Diodo: (a) simbolo; (b) sue caratteristiche

con una batteria interna (vedi per i dettagli: Noel M. Morris, *Elementi di elettronica teorica e pratica*, Hoepli). Quando l'anodo di un diodo efficiente è collegato al terminale "—" dello strumento e il catodo al terminale "+", la resistenza indicata dallo strumento dovrebbe essere bassa; quando i collegamenti sono scambiati lo strumento dovrebbe indicare una resistenza infinita.

Quando il diodo è polarizzato direttamente (Fig. 1.2b), la **caduta di tensione diretta** attraverso di esso varia generalmente da 0,2 a 0,6 V per un diodo al germanio e da 0,5 a 0,8 V per un diodo al silicio.

Quando il diodo è polarizzato inversamente con una tensione applicata inferiore alla tensione di rottura, si dice che funziona da **blocco**; in questo stato operativo soltanto la **corrente di perdita** fluisce attraverso il diodo. Nei diodi normalmente usati nei circuiti logici, il valore della cor-

rente di perdita è di solito di alcuni nanoampere (1 nA = 10^{-9}A); per le ragioni già viste la corrente di perdita aumenta leggermente con l'aumentare della temperatura.

Esiste un valore della tensione inversa noto come **tensione di rottura** (Fig. 1.2 b) al di là del quale la corrente in senso inverso aumenta rapidamente; a tensioni più elevate della tensione di rottura si dice che il diodo è in **conduzione inversa**. Nel caso di diodi usati nei circuiti logici e nei raddrizzatori, questo stato provoca un aumento della dispersione di potenza al loro interno che può raggiungere livelli pericolosamente alti e danneggiare perciò i diodi stessi. Diodi speciali noti come diodi Zener (vedi paragrafo 1.5) sono progettati per funzionare in conduzione inversa.

1.4 Diodi semiconduttori a giunzione p-n

Il diodo a giunzione p-n è costituito da un monocristallo di materiale semiconduttore contenente una regione anodica di tipo p e una regione catodica di tipo n (Fig. 1.3).

Quando un diodo è collegato come in Fig. 1.3(a), con l'anodo di tipo positivo rispetto al catodo di tipo n (cioè è polarizzato direttamente), si rileva un elevato passaggio di corrente nel circuito. La ragione è la seguente: in un materiale di tipo p i portatori di maggioranza sono buche e in materiali di tipo n sono elettroni.

Quando la regione p è collegata al polo positivo del generatore, questo inietta nell'anodo le buche mobili le quali sono attratte verso la regione n dal polo negativo del generatore. Nello stesso tempo, la regione n riceve nuovi elettroni dal polo negativo e queste cariche mobili sono attratte nella regione p dal polo positivo del generatore stesso. Il flusso di questi portatori di carica costituisce il flusso di corrente nel circuito. Da quanto precede risulta che **il diodo è polarizzato direttamente quando l'anodo di tipo p è positivo rispetto al catodo di tipo n**.

Se la polarità è invertita come in Fig. 1.3(b), la corrente nel circuito diventa praticamente nulla (rimane solo la corrente di perdita). In questo caso (polarizzazione inversa) il potenziale negativo applicato all'anodo di tipo p attrae le buche mobili (portatori di carica positiva) e le allontana dalla giunzione p-n. Nello stesso tempo, il potenziale positivo applicato al catodo di tipo n attrae gli elettroni mobili allontanandoli dalla giunzione p-n.

Il risultato complessivo è che non rimangono portatori mobili nella zona di giunzione p-n; cioè la zona attorno alla giunzione è privata dei
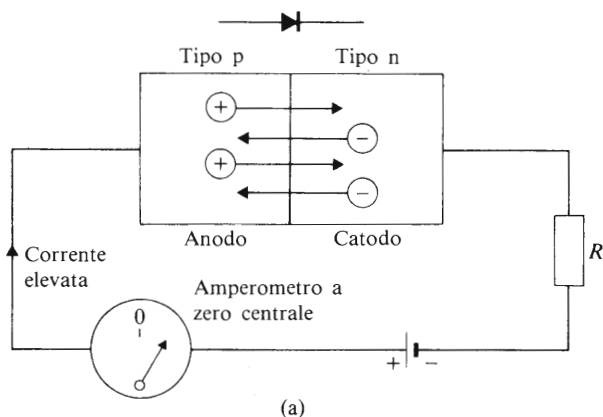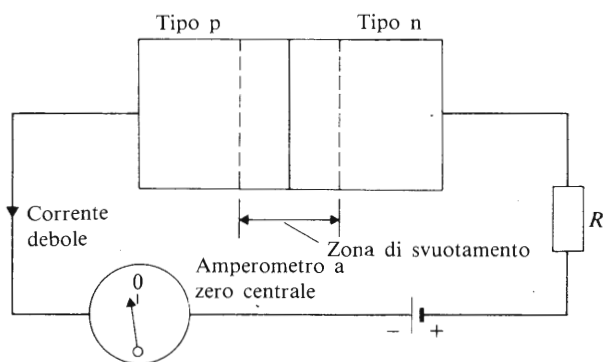
Fig. 1.3 Diodo a giunzione p-n polarizzato direttamente (a) e polarizzato inversamente (b)

portatori di carica ed è nota come **zona di svuotamento**. Dato che la zona di svuotamento non contiene portatori di carica liberi, diventa a tutti gli effetti uno strato isolante che impedisce ogni flusso di corrente attraverso il diodo, con l'eccezione della corrente di perdita. Nel caso di raddrizzatori, la rottura inversa del diodo avviene quando la tensione di polarizzazione inversa è sufficientemente elevata da rompere la zona di svuotamento. Nei diodi Zener (vedi paragrafo 1.5) la rottura inversa si ha invece per valori di tensione abbastanza piccoli.

1.5 I diodi Zener

Un diodo Zener è un diodo a giunzione p-n il cui drogaggio è molto più elevato di quello di un diodo normale, il che determina una rottura inversa per piccoli valori di tensione (si possono costruire diodi Zener con tensioni di rottura tra qualche volt e poche centinaia di volt). Per ogni singolo dispositivo, la tensione di rottura ha un valore stabile ragionevolmente costante in un campo abbastanza ampio di valori di corrente inversa.

I diodi Zener sono quasi sempre usati in polarizzazione inversa, con l'anodo collegato al polo negativo del generatore. Per ogni dispositivo Zener è definibile un valore di dissipazione massima di potenza. Facendo riferimento alla relazione

$$I = \frac{\text{dissipazione di potenza}}{\text{tensione di rottura}}$$

la I indicata è la massima corrente inversa che può circolare senza danneggiare il dispositivo. Per esempio, un diodo Zener da 4 V, 1 W, può sopportare una corrente inversa $I = 1/4 = 0{,}25$ A. Questi diodi trovano comune applicazione nei circuiti elettronici come sorgente di tensione di riferimento costante. Per questo tipo di applicazione, le pile campione sono state largamente soppiantate dai circuiti a diodi Zener.

Una semplice sorgente di riferimento è mostrata in Fig. 1.4. Abbiamo una tensione V_S non stabilizzata, fornita da un ponte di raddrizzamento e da un circuito di filtro. Il valore di V_S è generalmente il doppio del valore della tensione di rottura (V_Z) del diodo Zener. Il simbolo di diodo Zener è caratterizzato dal catodo piegato ad angolo (che simboleggia
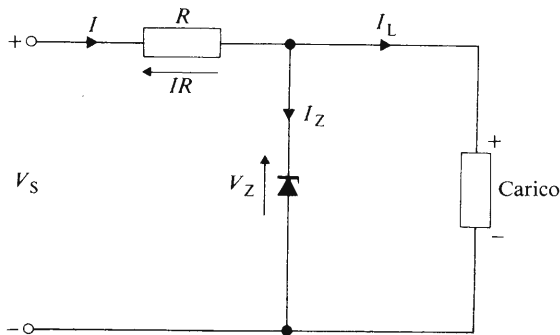
Fig. 1.4 Stabilizzatore di tensione con diodo Zener

l'effetto di rottura). Il resistore R assolve la doppia funzione di limitare la corrente e di determinare una caduta di tensione. La differenza dei potenziali V_S e V_Z appare ai capi di R. Dato che il potenziale di rottura di un diodo Zener è praticamente costante, la tensione ai capi del carico è pure costante, malgrado le variazioni della tensione fornita dal generatore; le variazioni di V_S appaiono solo ai capi di R e non ai capi del diodo Zener.

1.6 Tempo di immagazzinamento nei diodi

Per interdire un diodo a giunzione p-n che sta conducendo, è necessario prima eliminare tutte le cariche mobili nella regione della giunzione p-n. Solo dopo aver fatto questo, si può costruire nella zona della giunzione p-n la regione isolante di svuotamento. Il tempo perché questo si verifichi è detto *tempo di immagazzinamento* del dispositivo: esso, per i diodi usati in circuiti logici può variare da una frazione di nanosecondo (1ns = 10^{-9} s) a un millisecondo (1ms = 10^{-3} s) in un diodo ad alta corrente. Sebbene gli intervalli di tempo riportati siano piccoli, il tempo di immagazzinamento impone ugualmente un limite alla velocità massima con cui i dispositivi a giunzione possono essere utilizzati in commutazione. Quando un diodo porta una corrente elevata, è necessario un tempo superiore per rimuovere i portatori di carica e quindi il tempo di immagazzinamento cresce rispetto al valore assunto per le basse intensità di corrente.

Sebbene la descrizione che è stata data si riferisca ai diodi a giunzione, effetti di immagazzinamento di portatori di carica sono presenti in tutti i dispositivi a giunzione p-n (transitori, tiristori, triac, ecc.) quando essi passano dallo stato di conduzione a quello di interdizione.

1.7 Diodi a barriera Schottky

Un diodo Schottky è un dispositivo raddrizzatore a giunzione metallo-semiconduttore. Il materiale semiconduttore è solitamente di tipo n, ma possono essere usati anche materiali semiconduttori di tipo p. I diodi Schottky non presentano effetti di immagazzinamento di portatori di carica e hanno perciò un tempo di immagazzinamento nullo: questo per il diverso meccanismo di funzionamento dei diodi Schottky rispetto ai dispositivi a giunzione p-n.

Per ridurre il tempo totale di accensione o spegnimento di alcuni tipi di porte logiche, vengono inseriti nei loro circuiti alcuni diodi Schottky.

1.8 Transistori bipolari a giunzione

Il nome transistore è una contrazione di **TRAN**sfer re**SISTOR**, che gli inventori idearono per descrivere il suo funzionamento. Il termine "bipolare" proviene dal fatto che entrambe le polarità dei portatori di carica, cioè buche ed elettroni, sono coinvolte nel trasporto di corrente attraverso il dispositivo.

Il transistore bipolare è un dispositivo semiconduttore a 3 strati ottenuto da una fetta sottile (*chip*) di materiale semiconduttore, di solito silicio o germanio. Le tre regioni sono note come **emettitore, base** e **collettore** (Fig. 1.5).
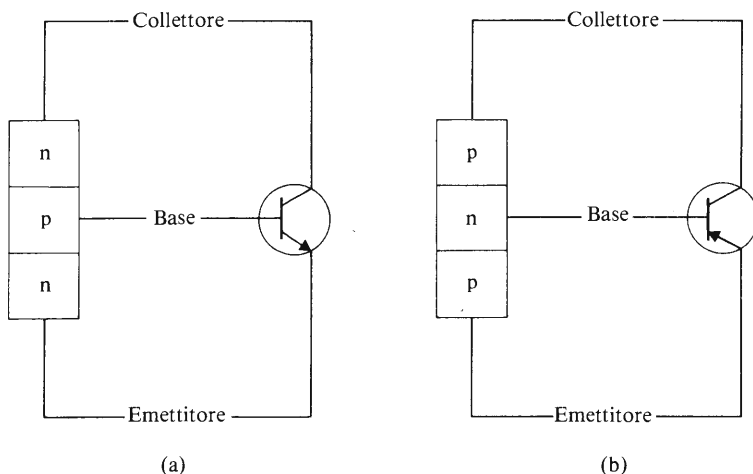
Fig. 1.5 Simboli per il transistore: (a) n-p-n; (b) p-n-p.

Il simbolo del transistore è disegnato in modo tale da fornire informazioni sul transistore stesso: una freccia è disegnata sempre sull'elettrodo emettitore, e indica il verso della corrente convenzionale (flusso di buche); in tal modo, se la freccia si allontana dalla base, la base è di tipo p; se la freccia si dirige verso la base, la base è di tipo n. L'emettitore deve il suo nome al fatto che da esso provengono i portatori di carica; il collettore invece è così chiamato perché la maggioranza dei portatori di carica che lasciano l'emettitore sono ivi raccolti. La sottile regione della base fornisce i mezzi elettrici per controllare il flusso di corrente attraverso il transistore; deve il suo nome al fatto che nei primi transistori era la pesante base fisica che faceva da supporto al transistore.

I due tipi principali di transistori a giunzione sono il tipo n-p-n e quello p-n-p (Fig. 1.5). Le dimensioni reali di un transistore sono molto ridotte; ad esempio un tipo di transistore ha una superficie di:

$$7 \times 30\ \mu m^2\ (1\ \mu m = 10^{-6}\ m)$$

Entrambi i tipi di transistori sono in uso e ciascuno di essi è costituito da un unico chip di materiale semiconduttore. Di seguito viene descritto un metodo di fabbricazione.

Nell'utilizzazione pratica, le sorgenti di tensione applicate a un transistore bipolare sono due: una alimenta il circuito di ingresso e l'altra il circuito di uscita, ed entrambe hanno un collegamento elettrico in comune. In un circuito reale le due tensioni menzionate sono fornite da uno stesso generatore.

Il tipo di elettrodo del transistore che è collegato al punto comune sopracitato caratterizza il circuito; abbiamo così il **circuito a emettitore comune**, il **circuito a base comune** e il **circuito a collettore comune**. Il circuito a emettitore comune è il più usato e una sua versione semplificata è mostrata in Fig. 1.6.

Quando tale circuito è usato come elemento logico di commutazione, la corrente di base I_B ha un valore molto elevato (elevato, s'intende, in termini elettronici) o è nulla. Consideriamo in primo luogo la condizione in cui I_B ha un valore elevato, cioè quando il valore della tensione dell'alimentatore di base V_{BB} polarizza direttamente la giunzione p-n emettitore-base.

Per studiare più in dettaglio il funzionamento del transistore è necessario sapere qualcosa di più sui livelli di drogaggio delle regioni al suo interno. Quando si costituiscono i transistori, si fa sempre in modo che la regione dell'emettitore sia molto drogata, cosicché possegga un'eccedenza di portatori mobili di carica; l'emettitore della Fig. 1.6 è di tipo n, cioè ha un eccesso di elettroni mobili. Inoltre, la base è drogata molto lievemente e dispone solo di pochi portatori mobili di carica (cioè di buche nella base di tipo p); difatti la base è drogata così poco da essere praticamente una regione di tipo i; essa è inoltre molto sottile (il suo spessore è dell'ordine della lunghezza d'onda della luce visibile).

Quando la giunzione emettitore-base è polarizzata direttamente, i portatori mobili di carica si muovono attraverso la giunzione stessa in entrambe le direzioni, ma il grande numero di elettroni emessi dall'emettitore verso la base è assai superiore a quello delle buche presenti nella base. Di conseguenza un'elevata concentrazione di elettroni (portatori di minoranza nella base di tipo p) viene a costituirsi nella base stessa. Le
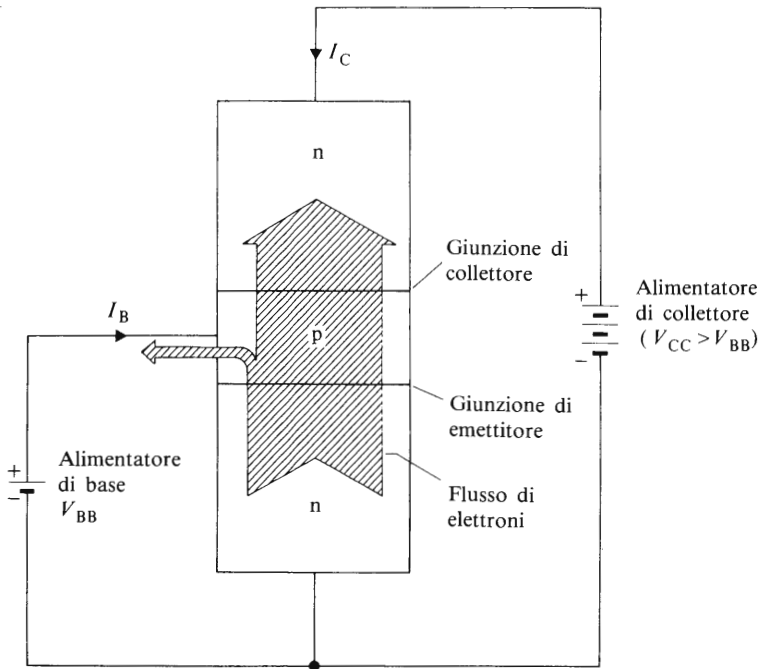
Fig. 1.6 - Collegamento con emettitore comune

buche mobili nella base che si combinano con alcuni di questi elettroni danno origine a un flusso di corrente (I_B) nel circuito di base esterno. Il valore del potenziale del collettore (V_{CC}) è normalmente superiore a quello del potenziale di base, cosicché la giunzione del collettore è polarizzata inversamente e un flusso di corrente normale in "direzione positiva" (cioè un flusso di buche) non può aver luogo tra di essi. Tuttavia la regione di base di tipo p ora contiene un'elevata concentrazione di elettroni (portatori di minoranza) ed essi vengono attratti verso il collettore dal potenziale positivo dell'alimentazione del collettore stesso. In pratica si trova che il 98 ÷ 99,9% degli elettroni che lasciano l'emettitore arrivano al collettore, e che molto pochi fluiscono attraverso il circuito di base esterno.

Il variare della corrente di base ha l'effetto di produrre una variazione quasi proporzionale nella corrente del collettore, e ciò viene usato per controllare il flusso di corrente attraverso il transistore. Quando la corrente di base è nulla, anche la corrente del collettore si annulla.

Così, il percorso tra il collettore e l'emettitore di un transistore può essere usato come un **interruttore elettronico** nel seguente modo: quando la corrente di base è elevata (tra i 10 e i 100 μA), la corrente del collettore è pure elevata e l'interruttore è ON, cioè chiuso; la resistenza del cammino tra il collettore e l'emettitore è bassa (circa 20 Ω). Quando la corrente di base è nulla, la corrente del collettore è pure nulla e l'interruttore è OFF, cioè aperto. Una semplice applicazione di questo genere di interruttori elettronici è descritta nel capitolo 3.

Nella televisione e nel campo delle radio ed audiofrequenze la corrente del transistore viene variata con continuità secondo l'intensità del segnale in arrivo, mentre nei circuiti logici la corrente può essere solo nello stato ON o nello stato OFF.

1.9 Transistori a effetto di campo (FET)

La variabile di controllo di un transistore a effetto di campo è una tensione che crea un campo elettrico all'interno del transistore stesso. In generale, i FET possono essere divisi in due grandi famiglie: quella dei **FET con il gate isolato** o **FET a Metallo-Ossido-Semiconduttore (JGFET o MOSFET)** e quella dei **FET con il gate a giunzione (JFET o JUGFET)**. I JUGFET hanno poche applicazioni nel campo dell'elettronica digitale e rimandiamo perciò il lettore al volume di Noel M. Morris, *Elementi di elettronica teorica e pratica*, Hoepli, per ulteriori informazioni su di essi.

1.10 I MOSFET

La sezione di un MOSFET è mostrata in Fig. 1.7(a). Il dispositivo in figura è formato da un substrato di materiale di tipo n; lo spessore del substrato è generalmente di 200 μm. Il dispositivo ha tre elettrodi detti **source** (pronuncia sōs), che è la sorgente dei portatori di carica immessi rispettivamente nel FET, **drain** (pronuncia dréin), dove i portatori di carica sono "scaricati" dal FET, e **gate** (pronuncia ghéit), che è l'elettrodo utilizzato per controllare il flusso di corrente attraverso il FET. I collegamenti tra il circuito esterno e gli elettrodi source e drain sono stabiliti mediante conduttori di alluminio che passano attraverso uno strato isolante di biossido di silicio (silice). Gli elettrodi source e drain sono materiali semiconduttori di tipo p che sono diffusi nel substrato di tipo n. Il gate è uno strato di alluminio isolato dal substrato tramite un sottilissimo velo di biossido di silicio; lo spessore di questo isolante è equivalente a 2 sole lunghezze d'onda della radiazione rossa!
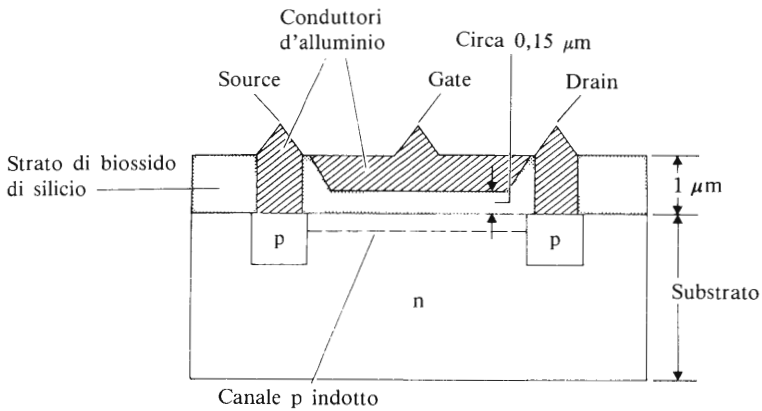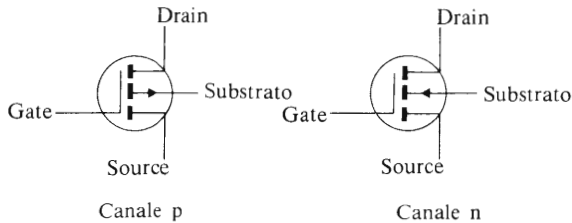
Fig. 1.7 (a) Sezione di un MOSFET a canale p. (b) Simbolo di un MOSFET a canale p. (c) Simbolo di un MOSFET a canale n

I lettori noteranno nella Fig. 1.7(a) che la zona tra il gate e il canale è formata di **M**etallo-**O**ssido-**S**emiconduttore, dalle cui iniziali deriva il nome MOSFET. Inoltre, dato che il gate è isolato dal substrato, questi transistori sono detti anche Insulated-Gate FET o IGFET.

Il dispositivo illustrato in Fig. 1.7(a) è noto come MOSFET a canale p, cosiddetto perché il flusso di corrente tra il source e il drain è dovuto al movimento di buche (cariche positive); in questo caso il source è collegato al polo positivo dell'alimentatore (sorgente perciò di buche) e il drain al polo negativo.

Quando la tensione tra il gate e il source è zero (cioè quando essi sono collegati tra di loro) la giunzione p-n tra il drain e il substrato è polarizzata inversamente (si ricordi che la regione del drain di tipo p è collegata

col polo negativo del generatore) e impedisce in questo modo il flusso di corrente. Di conseguenza, *quando la tensione di gate è zero la corrente del drain è zero*. Considerando il percorso tra il drain e il source come il contatto di un interruttore elettronico, l'interruttore è aperto o OFF quando la tensione di gate è zero.

Il substrato del MOSFET è molto poco drogato e, sebbene venga descritto come un materiale di tipo n, possiede un numero esiguo di elettroni mobili. Oltre a questi elettroni contiene una quantità addizionale di elettroni e buche generati da effetti termici secondo il processo descritto nel paragrafo 1.1. Se fosse possibile provocare l'accumulo in un unico punto delle buche generate termicamente, si avrebbe la conversione di una piccola zona del substrato lievemente drogato di tipo n in una zona di tipo p. Ciò può essere provocato applicando un potenziale negativo al gate (negativo rispetto al source); tale potenziale attrae le buche del substrato nella superficie inferiore dello strato di biossido di silicio appena sotto il gate. Per un valore del potenziale di gate noto come **potenziale di soglia**, un numero sufficiente di buche si raccoglie a formare un canale di conduzione tra il source e il drain: questo fenomeno crea il *canale di tipo p* della Fig. 1.7(a). Il valore del potenziale di soglia, detto valore di soglia V_T, per un dispositivo a canale di tipo p oscilla tra — 2 V e — 5 V. Portando il potenziale di gate a valori più negativi del valore di soglia, aumenta la conduttività del canale indotto e si determina una corrente di drain di maggiore intensità. Perciò l'interruttore elettronico è acceso quando viene applicato un potenziale di gate superiore al valore di soglia (in valore assoluto).

Un dispositivo di questo tipo è conosciuto come **FET a riempimento**, dato che aumentando (sempre in valore assoluto) il potenziale di gate aumenta la corrente di drain che provoca il riempimento del canale.

I simboli per il circuito della Fig. 1.7 forniscono informazioni sui MOSFET con il seguente schema: la linea tratteggiata tra il source e il drain implica che la corrente di drain è zero (il canale è interrotto) quando il potenziale di gate è zero; la distanza tra il gate e il canale indica che si tratta di un MOSFET; la direzione della freccia che parte dalla porzione centrale del canale indica il tipo di materiale del canale stesso: se la freccia è diretta *dal* canale *verso* il substrato (Fig. 1.7b) il canale è di tipo p, poiché la direzione della freccia è la stessa di quella associata a un diodo convenzionale di tipo p-n; se la freccia si dirige *verso* il canale (Fig. 1.7c), allora il dispositivo è un MOSFET a canale n.

Vengono prodotti sia MOSFET a canale n, sia a canale p. I dispositivi a canale n operano ovviamente con il drain collegato al polo positivo del generatore e con un potenziale di gate positivo.

1.11 Precauzioni nel maneggiare i MOSFET

Dato che lo strato di biossido di silicio tra il gate e il substrato è molto sottile, esso può venir permanentemente danneggiato dall'applicazione di tensioni relativamente basse nella regione del gate; il potenziale di rottura di questo strato di biossido di silicio varia fra 30 e 100 V. Tensioni di questo ordine di grandezza possono venire applicate semplicemente da chiunque maneggi il dispositivo; non è infrequente che una persona accumuli un potenziale statico in eccesso di 20 kV! I potenziali statici vengono generati dall'attrito tra la pelle e i vestiti o altri oggetti esterni. Per evitare possibili danni nel maneggiare e nel collaudare il dispositivo bisogna prendere alcune precauzioni:

1. Per evitare l'accumularsi di carica statica durante il trasporto e il contatto con le mani precedente l'uso effettivo del dispositivo, i suoi terminali dovrebbero essere in contatto con un materiale conduttore (per esempio una plastica conduttrice).
2. Tutti gli strumenti usati nel lavoro di montaggio dovrebbero essere collegati a terra; tra essi i saldatori e le parti metalliche dei supporti e degli strumenti.
3. I dispositivi MOS non dovrebbero essere inseriti o rimossi da circuiti quando è applicata una tensione esterna. Questo può provocare picchi di tensione nel circuito.
4. Quando i dispositivi MOS sono in uso, i terminali inutilizzati dovrebbero essere collegati a terra o alla linea di alimentazione, secondo ciò che sia conveniente per il tipo di porta logica in considerazione.
5. Non si dovrebbero applicare segnali alle linee di ingresso delle porte logiche mentre la tensione di alimentazione del dispositivo è nulla.
6. Le suddette precauzioni dovrebbero venir osservate nel collegamento con piastre che non siano inserite in un sistema in cui sono già applicate le corrette tensioni.

Altri metodi di protezione si possono fornire per l'impiego nel luogo dove i dispositivi MOS vengono montati sui circuiti. Essi includono appoggiapiedi conduttori sotto i tavoli di lavoro degli operatori, coprisgabelli conduttori, pavimenti conduttori, moquette conduttrici e ricopertura dei banchi pure conduttrice.

E' spesso consigliabile per gli operatori l'uso di un cinturino da polso messo a terra mentre lavorano con i dispositivi MOS, ma esso non deve essere considerato sufficiente come unico mezzo di protezione, anche perché è possibile dimenticarsi di collegarlo al circuito. Inoltre un cinturino di questo tipo potrebbe limitare i movimenti dell'operatore.

Un ulteriore pericolo può sorgere nel trattare gli elementi MOS se il dispositivo assemblato viene pulito con uno spray solvente dopo l'operazione di saldatura. La pulizia a spray può generare un'alta differenza di potenziale elettrico; bisogna consultare in proposito un esperto di montaggio.

Inoltre, è consigliabile per qualsiasi persona che debba maneggiare i dispositivi MOS portare una tuta fatta di materiale antistatico, anziché di nylon o di altra fibra sintetica.

1.12 Circuiti integrati monolitici (IC)

Dato che il silicio possiede caratteristiche che meglio si adattano ai processi di costruzione dei circuiti integrati, tutti i circuiti integrati monolitici sono realizzati con questa sostanza. La parola "monolitico" implica semplicemente che i circuiti integrati sono realizati da un monocristallo. I processi fondamentali di produzione sono desritti di seguito.

Innanzitutto, la silice (sabbia comune) è ridotta a silicio puro, dal quale si ricava un cristallo cilindrico avente le dimensioni tipiche di 30 cm di lunghezza e 7,5 cm di diametro. Il cristallo di silicio viene quindi tagliato da una sega diamantata in un gran numero di *wafer* cilindrici che, dopo la pulitura, hanno uno spessore di circa 200 μm, come illustrato nella Fig. 1.8(a) e (b). Dopo che il wafer di silicio è stato sottoposto a una serie di processi descritti in seguito, esso contiene un gran numero di singoli circuiti. Le dimensioni fisiche di questi circuiti possono essere anche molto piccole; in alcuni casi essi sono dei quadrati con il la-
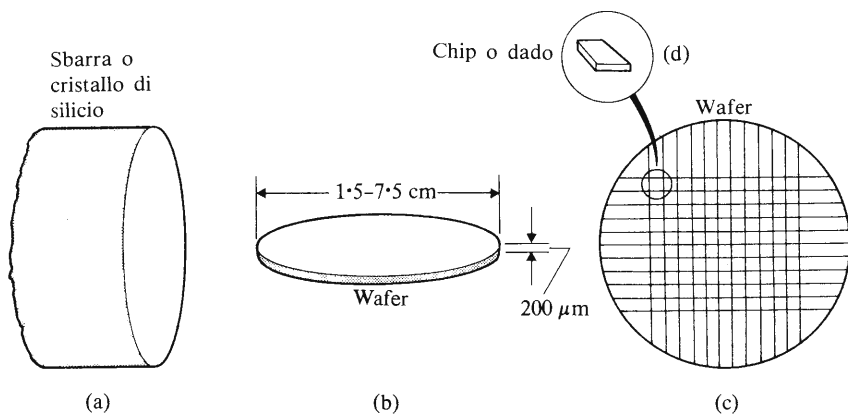
Fig. 1.8 Procedimenti per la costruzione di un circuito integrato monolitico

to lungo una frazione di millimetro. Per separare questi circuiti l'uno dall'altro, il wafer è diviso in tanti piccoli *chips* o *dadi* con procedimenti simili a quelli di taglio del vetro. Ciò è illustrato nei diagrammi (c) e (d) della Fig. 1.8. Ogni chip contiene un circuito completo.

Il chip viene quindi montato su un supporto e, dopo il completamento delle connessioni tra il circuito integrato e i terminali esterni, viene incapsulato per evitare contaminazioni dall'atmosfera circostante.

Il tipo di circuito integrato prodotto col metodo appena descritto è noto come circuito integrato *planare epitassiale* (epitassiale deriva dal greco e significa "costruito sopra"; l'espressione planare epitassiale significa che il circuito integrato è montato su una superficie piana).

1.13 Costruzione di un circuito integrato bipolare

Considereremo ora come viene costruito in forma integrata il circuito della Fig. 1.9(a); la rete mostrata potrebbe far parte di un circuito inter-
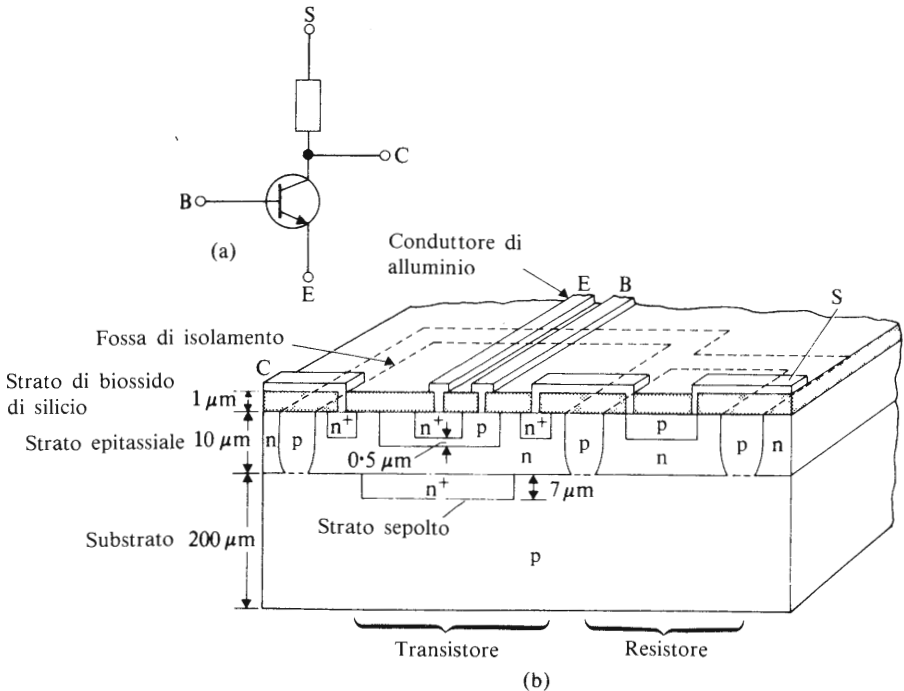
Fig. 1.9 Costruzione di un tipico circuito integrato monolitico

ruttore. Una volta completato, il circuito integrato appare come in Fig. 1.9(b), essendo le dimensioni totali del transistore di circa 100 μm^2. I passaggi intermedi del processo di costruzione sono riassunti di seguito.

Lo strato sepolto n^+

Il chip di silicio costruito nel processo iniziale è usato come *substrato* sul quale viene assemblato il circuito completo. La resistività del materiale del substrato è abbastanza elevata e ne deriverebbe che qualsiasi transistore costruito al suo interno sarebbe inutilizzabile come dispositivo interruttore. Per ovviare a questo difetto si diffonde nel substrato uno "strato sepolto" di tipo n^+ di materiale semiconduttore con un'elevata conduttività, in un punto sul chip che si trova direttamente sotto la zona dove il transistore si verrà a trovare alla fine. Un materiale di tipo n^+ è un qualsiasi materiale con un valore di conduttività superiore a quello di materiali convenzionali di tipo n. Il processo di diffusione è qui descritto sotto i titoli di *Crescita dell'ossido e mascheratura fotografica* e *Diffusione* ed è generalmente simile al processo di diffusione usato nella costruzione delle altre parti del circuito. Lo strato "sepolto" è così chiamato perché è effettivamente sepolto sotto la superficie del circuito.

Crescita dell'ossido e mascheratura fotografica

Dopo la pulitura e l'esame a vista, la superficie superiore del substrato è ossidata mediante passaggio di vapore su di essa; lo strato di ossido prodotto da questo processo ha uno spessore di circa 1 μm. La superficie superiore dell'ossido è ricoperta con un materiale fotosensibile noto come *photoresist*, come mostrato nella Fig. 1.10(a). Il photoresist è esposto a radiazioni ultraviolette attraverso una maschera fotografica (Fig. 1.10b) e le aree esposte del photoresist si induriscono. Le aree non esposte, coperte dalle aree opache della maschera, rimangono "soffici" e vengono dissolte con un solvente, lasciando un'apertura nel photoresist. La fetta è quindi attaccata chimicamente da un acido per rimuovere lo strato di ossido corrispondente all'apertura, lasciando così una "finestra" nella superficie superiore del substrato. Il photoresist residuo è poi rimosso mediante l'uso di altri solventi, dopo di che la fetta viene sciacquata ed essiccata.
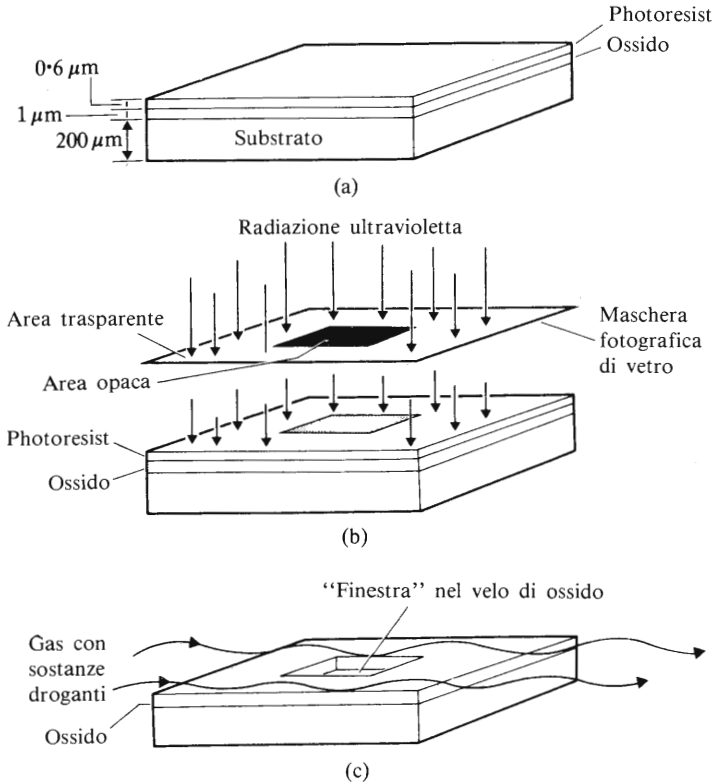
Fig. 1.10 Fabbricazione dello strato sepolto n^+

Diffusione

Nello stadio successivo della fabbricazione, la fetta viene fatta passare attraverso un forno di diffusione, dove viene scaldata a una temperatura di circa 1200°C e su di essa passano gas contenenti appropriate sostanze droganti (Fig. 1.10c). La sostanza drogante nel gas fa sì che l'area esposta del substrato di tipo p si converta in un materiale di tipo n^+. Lo strato sepolto, diffuso in questo modo attraverso la finestra nello strato di ossido, penetra fino a una profondità di circa 7 μm. Lo strato di ossido viene quindi asportato, lasciando il substrato di tipo p a contatto con lo strato sepolto n^+ sulla sua superficie.

Strato epitassiale

La fetta viene poi riscaldata un'altra volta in un forno e soggetta a gas che provocano la "crescita" di uno strato epitassiale uniforme su tutta la superficie (Fig. 1.11). In questo strato epitassiale, spesso 10 μm, viene costruito l'intero circuito integrato.
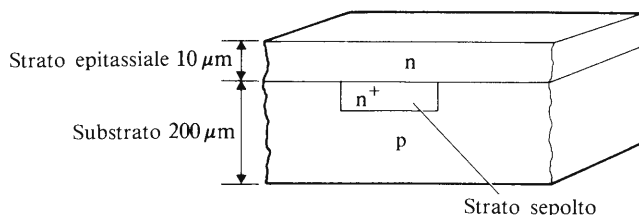
Fig. 1.11 Sezione trasversale di wafer dopo la formazione dello strato epitassiale

Componenti di circuito

Per isolare i componenti all'interno del circuito l'uno dall'altro, è necessario formare fosse di isolamento attorno alle zone in cui i componenti stessi verranno a formarsi. Con un processo di mascheratura, attacco chimico e diffusione simili a quelli sopra descritti, vengono diffuse le fosse di isolamento di tipo p all'interno dello strato epitassiale (Fig. 1.9b). La fossa di isolamento fornisce un collegamento tra la superficie del circuito integrato e il substrato e isola elettricamente l'area che essa circonda.

Vengono poi aperte finestre nello strato di ossido per potervi introdurre, mediante diffusione, la base di tipo p e il resistore. Il successivo processo di diffusione permette non solo l'introduzione dell'emettitore del transistore, ma anche delle due regioni n^+ nel collettore. Queste due regioni devono permettere sia una connessione con la regione del collettore vero e proprio che un'interconnessione tra il collettore e il resistore.

Completamento del circuito integrato

Uno strato di alluminio spesso circa 1,5 μm viene fatto evaporare sull'intera superficie del circuito e le regioni che non servono per le connessioni elettriche vengono rimosse tramite un processo di attacco chimico. Vengono quindi realizzate le connessioni tra l'alluminio rimasto e i piedini esterni del circuito integrato.

La descrizione appena riportata è una semplificazione dei processi adottati e, come il lettore noterà, anche la Fig. 1.9(b) è stata schematizzata. In totale, circa 80-100 singoli stadi sono necessari nella costruzione di un circuito integrato bipolare, alcuni dei quali richiedono poche ore, altri alcune settimane.

1.14 Costruzione dei circuiti integrati MOS

La sezione trasversale di un MOSFET a canale p costruito in forma integrata è riportata nella Fig. 1.7(a) e mostra le caratteristiche essenziali del dispositivo. Quando viene confrontata con il circuito bipolare integrato della Fig. 1.9, diventa subito evidente che il dispositivo MOS è più semplice da costruire. Inoltre richiede sul chip semiconduttore una superficie minore di quella di un transistore bipolare. Di conseguenza è possibile con gli elementi costitutivi di un MOS produrre un circuito più complesso su una data superficie del chip o produrre lo stesso circuito ad un costo inferiore che con gli elementi bipolari. In tal modo, le funzioni logiche nella maggior parte delle macchine calcolatrici elettroniche sono fornite da dispositivi MOS in forma di circuiti integrati.

1.15 Impaccamento di circuiti integrati

Le tre forme più comuni di impaccamento di circuiti integrati sono mostrate nella Fig. 1.12. La forma più consueta è il DIP (dual-in-line pack), di plastica incapsulata (Fig. 1.12a) con 14 piedini in doppia fila, sette per lato, ciascun paio allineato con gli altri a una distanza di circa 2,5 mm per permettere al circuito integrato di essere inserito direttamente su circuiti stampati standard. Il *canister* o *can* (Fig. 1.12b) porta il circuito integrato in un contenitore a forma di barattolo ermeticamente sigillato. Il *flatback* (Fig. 1.12c), spesso in ceramica, è pure sigillato ermeticamente.

1.16 Circuiti MSI e LSI

I termini *circuito a scala media di integrazione* (MSI = medium-scale integrated circuit) e *circuito a larga scala di integrazione* (LSI = large scale integrated circuit) sono usati generalmente per descrivere alcuni tipi di circuiti logici complessi. Si riferiscono al numero di porte logiche complete contenute in un integrato e, sebbene un po' imprecisamente, possono essere interpretati come segue:
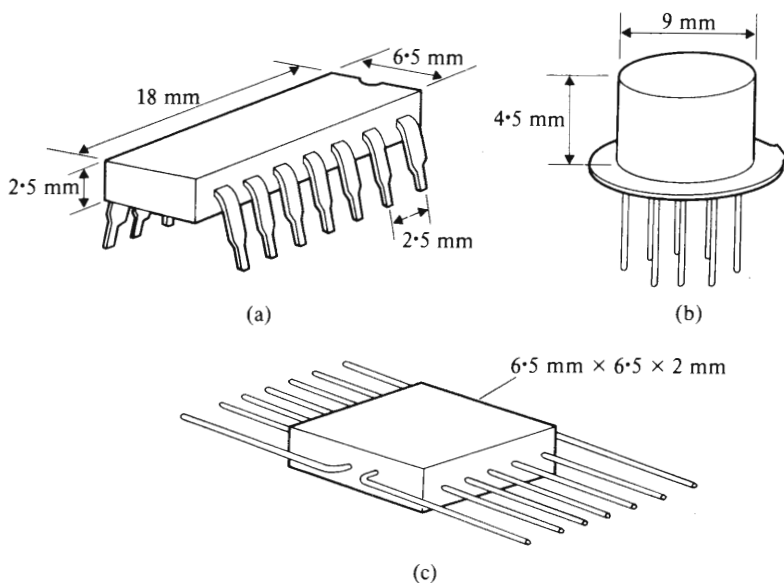
Fig. 1.12 Impaccamento dei circuiti integrati

a) i circuiti MSI contengono dalle 10 alle 100 porte logiche;
b) i circuiti LSI contengono più di 100 porte logiche.

I circuiti integrati usati nelle calcolatrici elettroniche sono circuiti LSI.

Capitolo 2

Porte logiche di base

2.1 Porte logiche

Un sistema logico è un sistema che funziona sempre in una maniera predeterminata, cosicché, dato un insieme di condizioni, si viene a produrre sempre lo stesso tipo di risposta.

Le regole basilari sono abbastanza facili da capire e da applicare: la funzione di una **porta logica** è quella di controllare il flusso di **informazioni** o di **dati** attraverso il sistema; il nome "porta" fu coniato in base al fatto che un dispositivo logico può presentarsi **aperto** al flusso di dati attraverso di esso, o **chiuso** rispetto allo stesso.

Bisogna sottolineare che il livello di potenza associato ad un segnale logico è molto basso (generalmente di pochi microwatt), ma il livello di potenza necessario per far funzionare la porta logica può essere relativamente elevato. Un'analogia tra i livelli di potenza che sono coinvolti in un'auto può aiutarci a illustrare la situazione. La potenza effettiva del segnale richiesta per far avanzare l'auto è semplicemente quella richiesta per inserire la marcia avanti (azione equivalente al nostro segnale logico), ma la potenza che il motore deve sviluppare per provocare il movimento in avanti dell'auto (equivale alla potenza richiesta per azionare la parte logica) può essere molto elevata. E'opportuno conoscere qualcosa sulle potenze in gioco in un sistema logico ma l'interesse dei lettori è diretto principalmente verso il controllo del flusso dei segnali logici attraverso il sistema.

Il funzionamento di ciascuna porta logica deve conformarsi a regole strettamente definite e le principali porte logiche sono denominate **AND, OR** e **NOT**; una descrizione delle versioni di queste porte è data in questo capitolo. Altre porte, note come **NAND** e **NOR**, sono descritte più avanti.

2.2 Livelli di segnali logici

Un sistema logico è un sistema che fornisce una soluzione definita per tutte le possibili condizioni in cui può trovarsi il circuito. Questo significa che l'uscita può sempre venir espressa nella forma di una risposta "sì" o "no" — la risposta non può mai essere "forse": abbiamo a che fare con un sistema a due livelli o **binario**, cosicché il potenziale di uscita da una porta logica può assumere solo 2 valori, per esempio +5 V o 0 V. Usando una notazione nota come **logica positiva**, il segnale di +5 V rappresenta la condizione di "sì" o "vero". In questa notazione un segnale "vero" è rappresentato dal cosiddetto "1" logico e un segnale "falso" dal cosiddetto "0" logico.

Si può usare anche una notazione alternativa, nota come **logica negativa**; in essa il segnale 0 V rappresenta l'"1" logico o il segnale "vero", mentre il +5 V rappresenta lo "0" logico, cioè il segnale "falso".

Sebbene abbiamo menzionato soltanto i livelli di potenziale di +5 V e 0 V, ogni differenza di potenziale come +3 V e —12 V può essere utilizzata in pratica. Nell'esempio appena citato avremmo:

	Esempio 1	*Esempio 2*
(a) con logica positiva:		
il potenziale massimo è "1"	+5 V	+3 V
il potenziale minimo è "0"	0 V	−12 V
(b) con logica negativa:		
il potenziale minimo è "1"	0 V	−12 V
il potenziale massimo è "0"	+5 V	+3 V

Sebbene la notazione della logica positiva sia quella più usata, la logica negativa è pure utilizzata in alcune famiglie logiche; si tratterà di questo più diffusamente nel paragrafo 2.9.

2.3 Un problema logico

Per illustrare il modo in cui operano i circuiti logici fondamentali, consideriamo il funzionamento di una ipotetica macchina per inscatolare giocattoli.

I giocattoli da inscatolare sono camion o automobiline; gli uni e le altre vengono trasportati su nastri scorrevoli indipendenti; se la scorta di scatole sta per esaurirsi, la macchina deve generare un segnale di allarme da trasmettere ai magazzini per indicare la penuria di scatole. La mac-

china dovrebbe continuare a operare finché c'è una scatola disponibile e se c'è un giocattolo da mettere nella scatola.

I paragrafi da 2.4 a 2.7 considerano il funzionamento dei circuiti che possono essere utilizzati per creare i segnali logici appropriati.

2.4 La porta NOT

Descrivendo i segnali logici, abbiamo visto che i segnali (ovvero le tensioni) in un sistema logico hanno solo due livelli possibili, e cioè quello corrispondente all'"1" logico e quello corrispondente allo "0" logico: ciò significa che, se l'uscita di una porta è "1" allora NOT è "0" e viceversa (in inglese NOT = non). Una porta NOT è una porta che fornisce uno "0" all'uscita quando il suo ingresso è "1", e un "1" all'uscita quando il suo ingresso è "0", cioè si dice che **inverte** o **complementa** il segnale logico che si presenta all'ingresso.

La Fig. 2.1 mostra un circuito NOT elementare che usa un relè. Una tensione viene applicata dall'interruttore B all'ingresso della porta (la bobina del relè) e l'uscita è prelevata attraverso i due contatti del relè. La resistenza R ha la funzione di limitare la corrente. Assumiamo che il circuito operi con la notazione della logica positiva, cioè che quando le tensioni di ingresso o di uscita sono $+E$ (positive), venga descritto con l'"1" logico e quando una tensione è zero venga descritta con "0" logico.

Nella Fig. 2.1 l'interruttore è aperto, $B = 0$ e la bobina del relè è smagnetizzata; i contatti del relè sono perciò aperti. In questo caso la corrente attraverso R e la caduta di potenziale attraverso di essa sono zero. La tensione di uscita è perciò $+E$ o l'"1" logico. Cioè il segnale
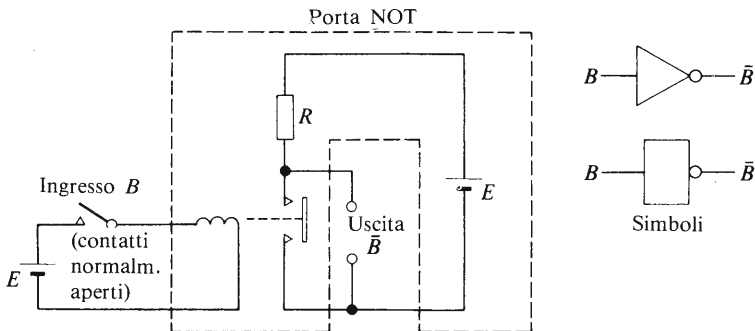
Fig. 2.1 Una porta logica NOT a relè

di uscita è il complemento logico del segnale d'entrata (ingresso = 0, uscita = 1).

Se l'interruttore è chiuso, il segnale $B = 1$ e la bobina del relè è magnetizzata; il contatto del relè è chiuso e la tensione di uscita è zero, ovvero lo "0" logico. Ancora, il segnale di uscita è il complemento logico del segnale d'ingresso (ingresso = 1, uscita = 0).

In pratica, il circuito a relè della Fig. 2.1 viene sostituito con un circuito semiconduttore (vedi capitolo 3). I simboli mostrati in figura possono essere utilizzati sia per il circuito a relè che per il suo equivalente a semiconduttore; questa osservazione è valida anche per tutte le altre porte logiche descritte nel capitolo.

Il funzionamento del circuito è riassunto in quella che viene chiamata la **tabella della verità** della porta (Tabella 2.1): essa è semplicemente un

Tabella 2.1 Tabella della verità per una porta NOT

Ingresso B	*Uscita* $\bar{B}$
0	1
1	0

metodo tabulare per mostrare le condizioni logiche di uscita del circuito per tutte le condizioni possibili di ingresso. Dato che alla porta NOT è applicato un solo segnale logico d'ingresso che può assumere una delle due condizioni (cioè "0" o "1"), ci sono solo due righe nella tabella della verità. L'uscita è espressa nella forma

$$\text{Uscita} = \text{NOT } B = \bar{B}$$

La lineetta è messa sopra B per significare l'inversione logica. ($\bar{B}$ si legge "B negato").

Consideriamo ora l'uso della porta NOT nel problema del paragrafo 2.3. Una condizione stabilisce che quando la scorta di scatole sta per esaurirsi, è necessario informare i magazzini affinché forniscano le scatole mancanti. Se le scatole sono impilate in una tramoggia e l'interruttore B è messo in qualche modo su di essa, quando la tramoggia contiene scatole l'uscita di B è l'"1" logico. L'uscita della porta NOT corrispondente al segnale di ingresso B può essere collegata a una lampada

d'allarme nei magazzini, la quale, quando si accende, illumina la scritta *carenza di scatole*. Chiaramente, quando ci sono scatole nella tramoggia allora $B=1$ e $\overline{B}=0$, quindi la lampada d'allarme è spenta. Quando la scorta di scatole si sta esaurendo, l'interruttore B si apre ($B=0$), $\overline{B}$ diventa 1 e la lampada **carenza di scatole** si accende.

2.5 La porta OR

Nella descrizione del sistema abbiamo detto che i giocattoli possono essere inscatolati se è presente un camion OR un'automobilina (in inglese OR = o, oppure). La presenza dei camion può essere indicata da un interruttore T e quella delle automobiline da un interruttore C. Se l'uscita della porta che decide se è presente almeno uno dei due tipi di veicoli è indicata con V, allora diciamo che

$$V=T \quad \text{OR} \quad C=T+C$$

dove il simbolo "più" (+) è usato per rappresentare la funzione logica OR. Il lettore non deve confondere il simbolo logico "+" con il simbolo di addizione aritmetica, poiché questi due simboli differiscono in un punto importante di cui parleremo più avanti.

In questo caso la porta ha due ingressi (da T e da C); quando T ha il valore "0", C può avere sia il valore "0" sia il valore "1". Anche quando T ha il valore "1", C può assumere il valore "0" o il valore "1". Ci sono perciò quattro combinazioni possibili delle due variabili; esse vengono elencate nella Tabella 2.2, che mostra inoltre l'uscita logica della porta OR per tutte le combinazioni possibili degli ingressi. La tabella della verità indica che quando $T=0$ e $C=0$, non ci sono veicoli

Tabella 2.2 Tabella della verità per una porta OR a 2 ingressi

Ingressi		*Uscita*
T	C	$V=T$ OR $C=T+C$
0	0	0
0	1	1
1	0	1
1	1	1

presenti e $V=0$. Quando c'è un'automobilina disponibile, ma non c'è un camion ($T=0$, $C=1$), possiamo riconoscere la presenza di un veicolo ($V=1$); quando c'è un camion disponibile, ma non c'è un'automobilina ($T=1$, $C=0$), di nuovo $V=1$. Anche quando sono disponibili sia un camion che un'automobilina, è di nuovo $V=1$. Così l'affermazione 1 OR $1=1+1=1$ è un'affermazione *logica* valida.

La tabella della verità della porta OR può essere riassunta come segue. **L'uscita è l'"1" logico se almeno uno degli ingressi è l'"1" logico; l'uscita è lo "0" logico solo quando tutti gli ingressi sono lo "0" logico.**

La Fig. 2.2 mostra una porta logica OR che contiene dei relè. I segnali di ingresso T e C sono ottenuti dagli interruttori nel modo esposto. Quando a entrambi gli ingressi T e C sono applicate tensioni nulle ($T=0$, $C=0$), entrambi i contatti del relè sono aperti e la tensione di uscita è 0 ($V=0$). Nel caso in cui una delle due bobine del relè (o anche entrambe) sia magnetizzata (quando $T=1$ OR $C=1$), il terminale di uscita è collegato al generatore interno della porta e $V=1$. Questo tipo di circuito perciò soddisfa la tabella della verità della porta OR.
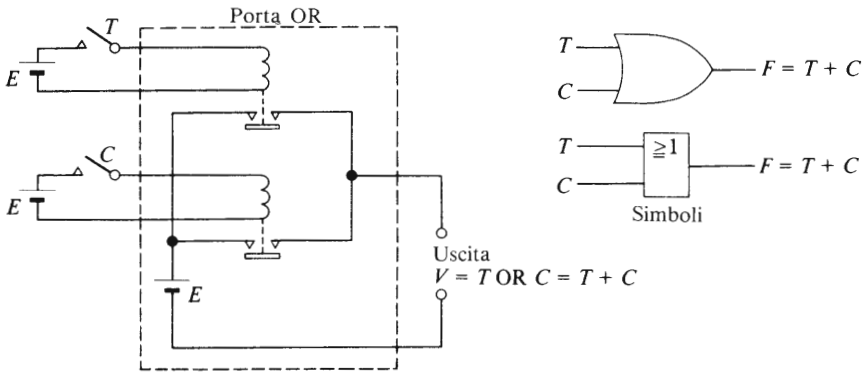
Fig. 2.2 Una porta OR

Nel caso più generale possono esserci numerosi ingressi nella porta, ciascuno guidato da un sensore indipendente e ciascuno dei quali alimenta una bobina indipendente. Supponiamo che gli ingressi provengano dai sensori A, B, C, D, E e che l'uscita sia designata con il simbolo X, allora $X = A$ OR B OR C OR D OR $E = A+B+C+D+E$

2.6 La porta AND

Nella macchina ipotetica descritta nel paragrafo 2.3, useremo una porta AND per far partire quella parte della macchina che mette un giocattolo nella sua scatola. Per inscatolare un giocattolo, la macchina deve simultaneamente avere a disposizione un veicolo (camion o automobilina) AND una scatola (in inglese AND = e). Se usiamo il simbolo S per rappresentare il segnale d'uscita della porta AND, allora si ha che:

$$S = V \text{ AND } B = V.B$$

dove il punto (.) è usato per denotare la funzione logica AND. Il lettore si ricorderà che il segnale V è generato da una porta OR (vedi paragrafo 2.5) e che il segnale B è ottenuto da un interruttore messo sulla pila di scatole.

Ancora una volta abbiamo due segnali d'ingresso o **variabili**, che danno quattro combinazioni di V e di B, come riportato nella Tabella 2.3. Dato che un veicolo può essere messo in una scatola solo quando la macchina ha a disposizione un veicolo e una scatola simultaneamente, allora $S=1$ solo quando $V=1$ AND $B=1$ simultaneamente (ultima riga della tabella 2.3). Quando almeno uno tra B e V assume il valore logico "0", $S=0$ (prime tre righe della tabella 2.3) e la macchina rimane inoperosa.

Tabella 2.3 Tabella della verità per una porta AND con 2 segnali d'ingresso

Ingressi		*Uscita*
V	B	$S = V$ AND $B = V.B$
0	0	0
0	1	0
1	0	0
1	1	1

La tabella della verità della porta logica AND può essere riassunta come segue. **L'uscita è lo "0" logico se uno degli ingressi (o entrambi) è lo "0" logico; l'uscita è l'"1" logico solo quando tutti gli ingressi sono l'"1" logico.**

La Fig. 2.3 mostra una porta logica AND a relè. Il lettore noterà che, se $V=0$ oppure $B=0$, la corrispondente coppia di contatti del relè è aperta e $S=0$. Solo quando $V=1$ AND (e) $B=1$ simultaneamente, sono chiuse entrambe le coppie di due ingressi alla stessa porta, ciascuno da un sensore indipendente. In generale si possono avere molti ingressi alla porta logica, ognuno dei quali è azionato da un sensore indipendente. Supponiamo che gli ingressi vengano dai sensori Q, R, S, T e che l'uscita sia contrassegnata dal simbolo Z, allora si ha

$$Z=Q \text{ AND } R \text{ AND } S \text{ AND } T=Q.R.S.T$$
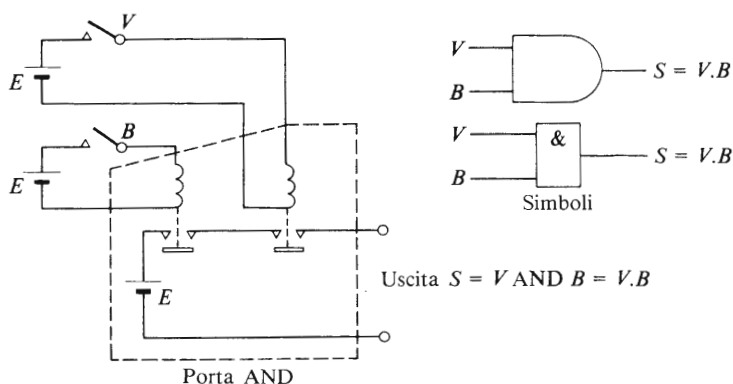
Fig. 2.3 Una porta AND

2.7 Il circuito completo di controllo

Un diagramma a blocchi del circuito logico che rappresenta il circuito di controllo della ipotetica macchina descritta nel paragrafo 2.3, è riportato nella Fig. 2.4.

Gli ingressi T e C sono ottenuti da sensori che rilevano la presenza dei corrispondenti tipi di veicolo, e il sensore B rileva la presenza o l'assenza di scatole per i giocattoli.

Il diagramma è tipico di un circuito logico nel senso che mostra solamente le connessioni necessarie per la trasmissione dei dati. Le altre connessioni, ad es. con i generatori, sebbene siano essenziali per il funzionamento del sistema, vengono omesse.

Poiché stiamo lavorando con uno schema logico a un livello relativa-
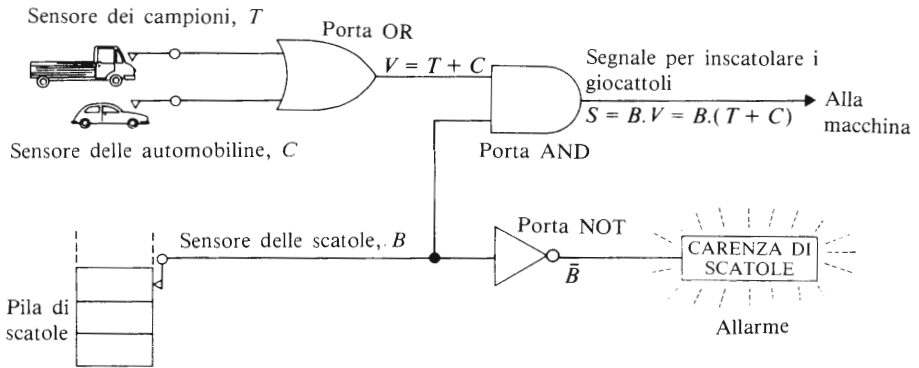
Fig. 2.4 Diagramma a blocchi del circuito di controllo per una macchina che inscatola giocattoli

mente elementare, il disegno della Fig. 2.4 mostra alcuni inconvenienti. Un problema abbastanza ovvio sorge quando un camion e un'automobilina arrivano simultaneamente; quale dei veicoli dovrà essere scelto? Questo e altri possibili difetti potranno essere corretti dal lettore stesso con l'aumentare delle sue informazioni ed esperienze sui sistemi logici.

2.8 Stesura di tabelle della verità

Una caratteristica comune a tutte le tabelle della verità è che, per un dato numero di linee di ingresso, il numero di variabili di ingresso rimane costante, cosicché per due variabili ci sono quattro combinazioni (vedi le Tabelle 2.2 e 2.3), per tre variabili ci sono otto combinazioni, per quattro variabili ci sono sedici combinazioni, ecc. Il numero di combinazioni può essere calcolato in base all'equazione:

$$\text{Numero di combinazioni} = 2^N$$

dove N è il numero delle variabili d'ingresso. Così, per tre variabili, $N = 3$ e ci sono $2^3 = 8$ combinazioni; per quattro variabili, $N = 4$ e si hanno $2^4 = 16$ combinazioni ecc.

Un modo in cui le combinazioni possono essere registrate nella tabella della verità è il seguente. Supponiamo di dover compilare una tabella della verità per una porta avente quattro variabili d'ingresso *A, B, C, D*

(ciascuna rivelata da un sensore nel modo già descritto). La colonna degli ingressi nella tabella della verità deve contenere tutte le sedici combinazioni delle quattro variabili. Nella Tabella 2.4 è riportata una tabella della verità per entrambe le porte OR e AND con quattro linee di ingresso. Il metodo adottato dall'autore è quello di cominciare a scrivere sotto la variabile A una serie di otto 0 (questo numero di zeri è uguale alla metà del numero totale delle combinazioni d'ingresso, cioè $16/2 = 8$) seguito da otto 1 per completare le 16 file.

Tabella 2.4 Tabella della verità per una porta con 4 segnali d'ingresso

Ingressi				*Uscita*	
A	*B*	*C*	*D*	*Porta AND*	*Porta OR*
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	1
0	0	1	1	0	1
0	1	0	0	0	1
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	0	1
1	0	0	0	0	1
1	0	0	1	0	1
1	0	1	0	0	1
1	0	1	1	0	1
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	1
1	1	1	1	1	1

Sotto la variabile B scriviamo quattro 0 (pari alla metà degli zeri che compaiono nella colonna A), seguiti da quattro 1; questo viene ripetuto fino a completare le sedici righe della colonna B. La colonna C è compilata scrivendo due 0 (pari alla metà degli zeri che compaiono nella colonna B), e due 1; questa sequenza di 0 e di 1 va ripetuta fino a completare la colonna C. La colonna D è completata scrivendo una sequenza di 0 e di 1 alternati.

Se le variabili d'ingresso fossero cinque ci sarebbero $2^5 = 32$ combinazioni di queste variabili. La prima colonna d'ingresso della tabella del-

la verità conterrebbe una sequenza di sedici 0 e sedici 1, la seconda colonna otto 0 seguiti da otto 1 (ripetuti due volte), la terza colonna quattro 0 seguiti da quattro 1 (ripetuti quattro volte), ecc. fino a che tutte le colonne fossero complete.

Gli 0 e gli 1 nella colonna di uscita dipendono dal tipo di porta o dal tipo di connessione logica che lega l'entrata all'uscita. Nel caso di una porta AND a quattro ingressi, l'uscita è "1" logico soltanto quando tutti i quattro segnali di ingresso sono "1" logico simultaneamente (ultima riga della tabella di verità). Nel caso di una porta OR, l'uscita è lo "0" logico soltanto quando tutti i segnali di entrata sono lo "0" logico simultaneamente.

2.9 L'effetto della notazione logica positiva e negativa sulle porte logiche

Abbiamo già visto in questo capitolo, che un particolare valore di tensione può rappresentare l'"1" logico o lo "0" logico secondo che si usi una notazione logica positiva o negativa. Per illustrare l'effetto di una notazione logica sul "nome" dato alla porta logica, esaminiamo ora un caso particolare.

Supponiamo che le tensioni misurate agli ingressi e all'uscita di una porta logica siano quelle riportate nella Tabella 2.5. Consideriamo ora le tabelle della verità che derivano dall'uso della notazione logica positiva e di quella negativa.

Operazione logica positiva

In questa notazione la tensione più alta (4 V) corrisponde all'"1" logico e la tensione più bassa (0,5 V) allo "0" logico. Riscrivendo i valori di tensione della Tabella 2.5 nei termini della notazione logica positiva, otteniamo la Tabella 2.6.

Confrontando la Tabella 2.6 con la Tabella 2.2 per una porta OR a due ingressi, notiamo che in questo caso la funzione logica generata dalla porta è la funzione OR. Cioè

$$F = A \quad \text{OR} \quad B = A + B$$

Tabella 2.5

Ingressi (volt)		*Uscita (volt)*
A	*B*	*F*
0,5	0,5	0,5
0,5	4,0	4,0
4,0	0,5	4,0
4,0	4,0	4,0

Tabella 2.6

Ingressi (volt)		*Uscita*
A	*B*	*F*
0	0	0
0	1	1
1	0	1
1	1	1

Tabella 2.7

Ingressi (volt)		*Uscita*
A	*B*	*F*
1	1	1
1	0	0
0	1	0
0	0	0

Operazione logica negativa

Nella notazione logica negativa la tensione più alta rappresenta lo "0" logico e la tensione più bassa l'"1" logico. Riscrivendo i valori di tensione della Tabella 2.5 nei termini della notazione logica negativa, otteniamo la Tabella 2.7.

Confrontando la Tabella 2.7 con la Tabella 2.3 per una porta AND a due ingressi (il lettore noterà che ora le combinazioni vengono riportate in una sequenza diversa), vediamo che la porta funziona come una porta AND. Cioè

$$F = A \quad \text{AND} \quad B = A.B$$

Conclusione

Da quanto riportato è evidente come il nome che diamo a una porta logica (definita dalla sua tabella della verità) dipenda dalla notazione logica usata. Questo non deve preoccupare il lettore, poiché la funzione logica esplicata dalla porta, cioè AND oppure OR, è specificata dal costruttore (ciò significa che il suo uso con una delle due notazioni positiva o negativa è già stato determinato). Tuttavia il lettore dovrebbe rendersi conto che un dato circuito operante a specifici livelli di tensione (4 V e 0,5 V nel caso citato) può trovarsi in un tipo di costruzione a generare la funzione AND, e in un altro tipo di costruzione a generare la funzione OR.

La maggior parte delle porte considerate in questo libro opera con logica positiva. Verranno fornite indicazioni specifiche per le porte che operano invece in logica negativa.

Capitolo 3

I circuiti logici

In questo capitolo si introduce il lettore alla conoscenza di molteplici porte logiche elettroniche.

3.1 Una porta logica OR a diodo-resistore (DRL)

La Fig. 3.1 mostra una comune porta OR DRL. Ciascuna linea di ingresso contiene un diodo, ciascun diodo agisce come un interruttore elettronico sensibile alla tensione: esso è chiuso quando il diodo è polarizzato direttamente, ed è aperto quando la tensione ai suoi capi è zero o quando è polarizzato inversamente.
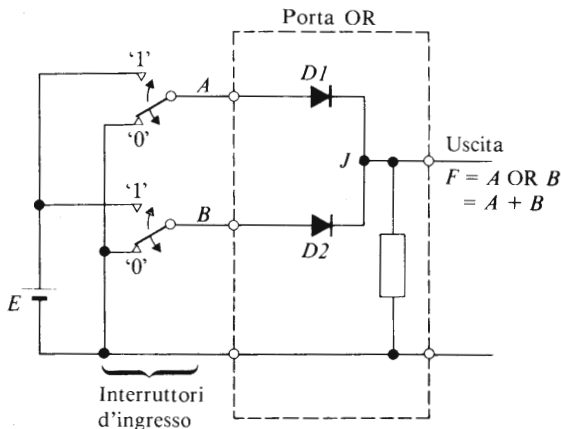
Fig. 3.1 Una porta logica OR a diodo resistore

Si suppone nel seguito che il circuito operi con una logica positiva, cioè *E* volt sono equivalenti all'"1" logico e 0 volt allo "0" logico. Quando gli interruttori di ingresso sono nella posizione della Fig. 3.1,

$A = 0$ e $B = 0$; in questo caso una tensione nulla è applicata agli ingressi e perciò l'uscita è zero ($F = 0$). Questo corrisponde alla prima riga della Tabella 3.1. Quando un "1" logico è applicato a uno dei due ingressi mentre l'altro ingresso è zero (cioè $A = 1$, $B = 0$ oppure $A = 0$, $B = 1$ in corrispondenza alle righe 2 e 3 della tabella della verità), il diodo nella linea con l'"1" logico è polarizzato direttamente, e ciò produce come tensione di uscita E volt ($F = 1$).

Se il segnale logico 1 è applicato a entrambe le linee d'ingresso contemporaneamente ($A = 1$, $B = 1$), entrambi i diodi sono polarizzati direttamente e $F = 1$ (ultima riga della tabella della verità).

Tabella 3.1 Tabella della verità per una porta OR a due ingressi

Ingressi		*Uscita*
A	B	$F = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

Il circuito della Fig. 3.1 è molto semplice. Il suo principale inconveniente è la caduta di tensione nei diodi quando sono polarizzati direttamente. Se soltanto una o due porte di questo tipo sono collegate in serie (o in cascata) l'una con l'altra, la caduta di tensione totale può essere accettabile. Se vengono comprese molte porte, allora la caduta di tensione totale attraverso i diodi può far sì che la tensione di uscita finale corrispondente a un segnale di ingresso "1" logico cada al di sotto un livello accettabile all'uscita del sistema logico.

Nel circuito considerato, il numero dei segnali d'ingresso (variabili d'ingresso) è limitato a due, poiché ci sono soltanto due diodi. Tuttavia, se vogliamo aumentare il numero di linee d'ingresso (il fan-in della porta — vedi anche il paragrafo 3.2), possiamo collegare altre linee di ingresso mediante diodi col catodo accoppiato al nodo J.

3.2 Fan-in e fan-out

Il **fan-in** di una porta logica è il numero massimo di segnali indipendenti d'ingresso che può essere connesso con la porta stessa senza che essa funzioni in modo errato. Per funzionamento errato intendiamo che il segnale di uscita scenda al di sotto del livello dell'"1" logico quando dovrebbe essere l'"1" logico, oppure salga al di sopra dello "0" logico quando dovrebbe essere lo "0" logico.

Il progettista di porte logiche deve preoccuparsi del numero massimo di linee d'ingresso che può fornire all'utilizzatore; egli si trova in una posizione difficile, poiché deve considerare molti fattori, tra cui il costo finale della porta (l'utilizzatore non vuole comprare prestazioni di cui non ha bisogno!), la velocità operativa della porta (che è in qualche modo dipendente dal numero delle linee d'ingresso), il consumo di potenza (che è indirettamente correlato col numero di linee d'ingresso), ecc. L'utilizzatore è in ogni caso interessato al numero di linee d'ingresso che può sfruttare. Se il fan-in disponibile (che è due nel caso della Fig. 3.1) è insufficiente, può essere necessario ridisegnare la rete logica. Alternativamente è possibile per alcune famiglie di porte impiegare un procedimento di espansione del fan-in, che permette di aumentare di un fattore dieci o venti la capacità di fan-in di una porta tipo quella della Fig. 3.1. D'altra parte, se una porta ha più linee d'ingresso di quante ne occorrano all'utilizzatore, che cosa si deve fare? Quest'ultimo problema verrà discusso nel paragrafo 3.5.

Il **fan-out** di una porta logica è il numero di porte logiche che può essere connesso all'uscita della porta senza provocare il mal funzionamento di questa.

Per prevenire i problemi che sorgono quando si usa una famiglia logica particolare, si consiglia al lettore di raccogliere il maggior numero di informazioni possibile circa la famiglia stessa. Per esempio, ogni ingresso in una porta TTL (vedi paragrafo 4.3) rappresenta un'*unità di carico* per la parte pilotante.

Il fan-out di ciascuna porta è generalmente specificato dal costruttore e, per esempio, nel caso dell'integrato NAND a due ingressi SN 7400 TTL (Fig. 4.8), ciascuna porta ha un fan-out di dieci unità di carico; questo significa che può essere connessa a 10 ingressi di altrettante porte TTL. Tuttavia, nel caso di alcuni tipi di porte TTL, ciascuna linea di ingresso può presentare un carico superiore all'unità.

Il fan-in, il fan-out e la velocità di commutazione delle porte logiche sono generalmente correlati, e i costruttori spesso forniscono tabelle che mostrano gli effetti reciproci tra questi parametri.

3.3. Una porta logica AND a diodo-resistore

Un tipo di porta logica AND a diodo-resistore è mostrato nella Fig. 3.2, nella quale E_1 ed E_2 hanno lo stesso valore (spesso sono presi direttamente dallo stesso generatore). La tabella della verità è riportata nella Tabella 3.2.

Quando gli interruttori d'ingresso sono nella posizione della figura, i segnali logici d'ingresso sono $A = 0$ e $B = 0$ (prima riga della tabella della verità). A questo punto i catodi di entrambi i diodi sono connessi con la linea di zero volt; dato che entrambi gli anodi sono connessi al
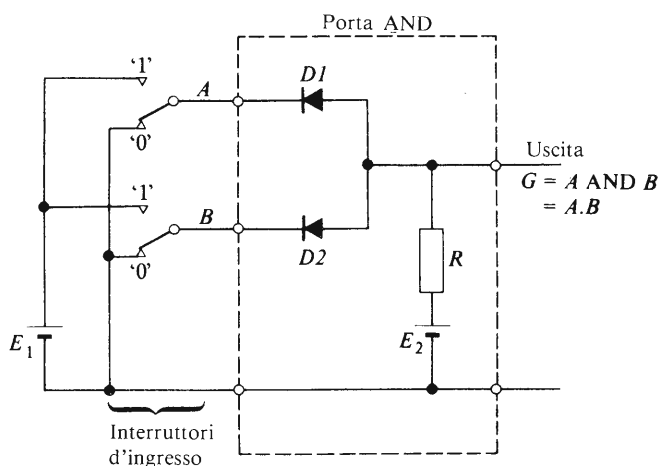
Fig. 3.2 Una porta logica AND a diodo-resistore

Tabella 3.2 Tabella della verità per una porta AND a due ingressi

Ingressi		*Uscita*
A	B	$G = A$ AND(e) B - $= A.B$
0	0	0
0	1	0
1	0	0
1	1	1

potenziale $+E_2$ tramite il resistore R, entrambi i diodi sono polarizzati direttamente. Se supponiamo per il momento che i diodi siano "ideali", cosicché la caduta di tensione attraverso di essi sia zero quando essi sono polarizzati direttamente, allora nelle condizioni su esposte la tensione di uscita è uguale alla tensione attraverso i diodi, cioè zero volt. Così quando $A = 0$ e (AND) $B = 0$, allora $G = 0$. In pratica, quando un diodo conduce, appare ai suoi estremi una differenza di potenziale (circa 0,6 V in un diodo al silicio), cosicché lo "0" logico in uscita ha un valore più elevato di zero volt.

Supponiamo di cambiare il segnale di ingresso in B in un "1" logico ($A = 0$, $B = 1$), corrispondente alla seconda riga della tabella della verità. Dato che il catodo del diodo *D1* rimane a potenziale zero, esso è tuttora polarizzato direttamente dalla tensione E_2 e l'uscita della porta rimane lo "0" logico, cioè $G = 0$. Nello stesso tempo, dato che l'anodo del diodo *D2* è collegato all'uscita (che è a zero volt circa) e il suo catodo è connesso al potenziale $+E_1$ attraverso l'interruttore d'ingresso, il diodo *D2* è polarizzato inversamente e non prende parte al funzionamento del circuito.

Se cambiamo posizione agli interruttori d'ingresso in modo che $A = 1$ e $B = 0$ (terza riga della tabella della verità), vediamo, con un argomento simile a quello del paragrafo precedente, che il diodo *D2* è polarizzato direttamente e che $G = 0$. In queste condizioni il diodo *D1* è polarizzato inversamente e non prende parte al funzionamento del circuito.

Infine, quando mettiamo A e (AND) B nello stato di "1" logico (ultima riga della tabella della verità), entrambi i diodi *D1* e *D2* sono a tutti gli effetti circuiti aperti. In questo caso non passa corrente attraverso la resistenza R e la caduta di tensione attraverso di essa è zero. Quindi la tensione di uscita è uguale a E_2 (che corrisponde all'"1" logico). In pratica una piccola corrente viene sottratta dai terminali di uscita dal carico ad essi connesso, cioè dalle porte ad essi collegate, cosicché la tensione dell'uscita sarà minore di E_2.

3.4. Porte "generatrici" di corrente e porte "utilizzatrici" di corrente

I termini logica generatrice di corrente e logica utilizzatrice di corrente vengono frequentemente usati nella descrizione di sistemi logici. Gli elementi essenziali di entrambi i tipi sono illustrati nella Fig. 3.3.

Una porta logica **che genera corrente** (Fig. 3.3a) è una porta in cui l'uscita della porta pilotante agisce come generatore di corrente per l'ingresso della porta pilotata. La porta logica OR a diodo-resistore (Fig.
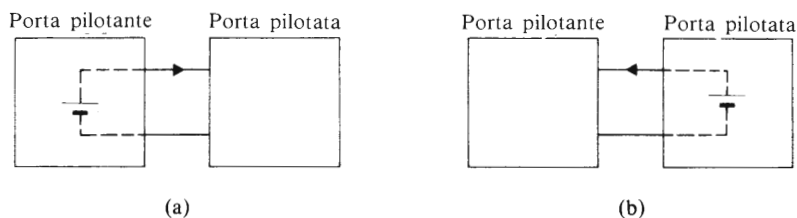
Fig. 3.3 (a) Sistema logico generatore di corrente; (b) sistema logico utilizzatore di corrente

3.1) è un esempio di porta che genera corrente. Per illustrare il principio generale, consideriamo la porta logica OR a diodo-resistore in Fig. 3.1 come una porta pilotata, con gli interruttori d'ingresso sostituiti dai segnali di uscita dalle porte pilotanti: allora vediamo che le porte pilotanti devono fornire corrente attraverso i diodi *D1* e *D2* per fare in modo che la porta pilotata produca un segnale in uscita.

Una porta logica **che utilizza corrente** (Fig. 3.3b) è una porta nella quale la porta pilotante deve utilizzare o assorbire la corrente che esce dai terminali d'ingresso della porta pilotata. La porta logica AND a diodo-resistore della Fig. 3.2 è un esempio di questo genere di porta. Nel caso della Fig. 3.2, se consideriamo la porta AND come la porta pilotata e ciascun interruttore come l'uscita di una porta pilotante, quando il segnale logico all'uscita della porta pilotante è lo "0" logico, questa deve assorbire la corrente che fluisce attraverso la linea d'ingresso della porta pilotata. Così l'interruttore della linea *A* nella Fig. 3.2 deve assorbire la corrente che esce dal diodo *D1*.

3.5 Il rumore nei sistemi logici

Il rumore elettrico può essere definito come ogni tipo di tensione che non fa parte dell'informazione d'ingresso. Il rumore può comparire in quasi tutti i conduttori del sistema e può essere dovuto a un gran numero di cause, che comprendono interferenze elettrostatiche e elettromagnetiche. Può essere generato all'interno del sistema logico stesso, e in questo caso è dovuto all'accensione e allo spegnimento delle porte, e può essere prodotto esternamente dall'accensione o dallo spegnimento dei dispositivi elettrici. Alcuni effetti del rumore elettrico sui circuiti logici sono descritti qui di seguito.

Segnali di rumore possono comparire sulle linee di ingresso o su ciascuna delle linee di alimentazione. Il risultato complessivo è lo stesso, in

quanto il segnale di rumore può causare un cattivo funzionamento del circuito; in generale è desiderabile che le porte logiche possiedano un alto grado di **immunità al rumore**. Per migliorare l'immunità al rumore di una porta particolare può essere necessario, a livello di progetto, cercare compromessi con alcune caratteristiche come la velocità di commutazione, il consumo di potenza, la capacità di fan-in o di fan-out, ecc.

Consideriamo ora il caso di una porta logica OR a diodo-resistore come quella della Fig. 3.1. Se entrambe le linee di ingresso hanno applicati dei segnali logici "0", allora l'uscita è lo "0" logico; qualora il sistema non sia sufficientemente schermato dal punto di vista elettrostatico ed elettromagnetico, è possibile che, se nelle vicinanze viene avviato un motore elettrico di grande potenza, un picco di tensione venga indotto in una delle linee dei segnali d'ingresso. Un picco di rumore in senso positivo potrebbe far sì che la porta produca momentaneamente un segnale logico "1" di breve durata al suo terminale di uscita; questo segnale potrebbe provocare una serie di altre operazioni sbagliate lungo il circuito logico. E' perciò essenziale seguire le raccomandazioni del costruttore riguardo alla protezione e alla schermatura non solo delle linee di connessione delle porte logiche, ma anche del sistema logico e delle sue alimentazioni.

E' inoltre possibile avere tensioni di rumore indotte anche sulle linee di ingresso non utilizzate di una porta logica. Una soluzione frequentemente adottata in questo caso è quella di collegare tutte le linee d'ingresso non utilizzate a terminali che non hanno effetto sul funzionamento del circuito (questo equivale a dire che tutte le linee d'ingresso devono venire collegate con qualche punto del circuito).

Consideriamo per esempio la porta logica OR a diodo-resistore della Fig. 3.1. Uno "0" logico in una qualsiasi delle linee d'ingresso non ha alcuna influenza sul segnale di uscita, cosicché è possibile collegare tutte le linee d'ingresso non utilizzate a un segnale logico "0", come mostrato nella Fig. 3.4(a). In alternativa, ciascuna linea d'ingresso non utilizzata può essere collegata a un'altra linea di ingresso utilizzata, come illustrato nella Fig. 3.4(b).

Nella porta logica AND a diodo-resistore della Fig. 3.2, un "1" logico su una qualsiasi delle linee di ingresso fa sì che il diodo di ingresso su quella linea venga polarizzato inversamente; quando accade questo, la linea d'ingresso in questione non prende più parte al funzionamento del circuito, ma non inibisce il funzionamento della porta.

Una disposizione conveniente di porte AND (e di porte NAND, come vedremo più avanti) è quella di connettere tutte le linee d'ingresso non utilizzate alla linea che porta il segnale logico "1", come mostrato nella

Fig. 3.4(c). In alternativa, le linee non utilizzate possono essere collegate alle linee d'ingresso utilizzate (Fig. 3.4d).
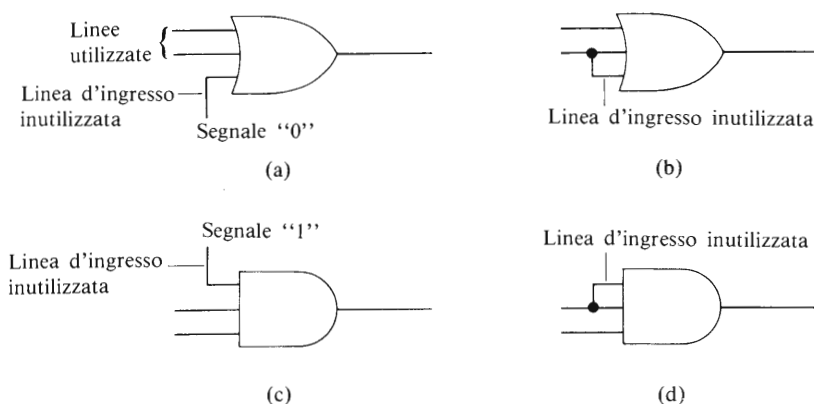
Fig. 3.4 Metodi di trattamento delle linee di ingresso inutilizzate

3.6 Il transistore come interruttore — La porta NOT

Nella Fig. 3.5 è disegnato un transistore equivalente a una porta NOT a relè (vedi capitolo 2). Il transistore funziona come un interruttore elettronico, il segnale di uscita viene raccolto ai capi del "contatto" dell'interruttore (lungo il cammino dal collettore all'emettitore). Dato che il circuito della Fig. 3.5 contiene soltanto resistori e un transistore, viene descritto come una porta logica NOT a resistore-transistore (RTL = resistor-transistor logic). Questo tipo di circuito rappresentò la base di una delle prime famiglie logiche, ma è stato superato dagli sviluppi tecnologici.

Quando l'interruttore S è nella posizione indicata, la tensione $V_1 = 0$, cosicché la corrente di base nel transistore è pure zero. Come già visto nel capitolo 1, anche la corrente nel collettore è zero; il risultato complessivo è che la differenza di potenziale ai capi di R_L è zero, cosicché la tensione di uscita è +9 V. Supponendo che la porta lavori con una notazione logica positiva, le condizioni di funzionamento appena descritte significano che un ingresso corrispondente alla "0" logico produce un'uscita corrispondente all'"1" logico.

Quando l'interruttore S è spostato nella posizione superiore, cosicché $V_1 = 9$ V (il che corrisponde a un ingresso di un "1" logico), la corren-
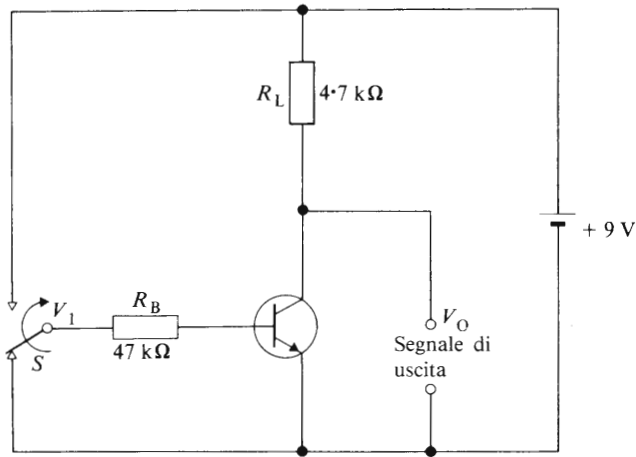
Fig. 3.5 Un'elementare porta logica a transistore-resistore NOT (RTL)

te di base è elevata (elevata in termini di correnti elettroniche) e il transistore conduce fortemente. In questa condizione la differenza di potenziale ai capi del transistore si riduce a un valore molto basso (di solito compreso tra 0,1 e 0,3 V), che corrisponde a un'uscita di uno "0" logico. Così un segnale logico di ingresso "1" dà origine a un segnale logico di uscita "0".

3.7 Le porte logiche NOT a MOS

Due tipi di porte logiche NOT a MOSFET sono illustrate nella Fig. 3.6. La porta NOT della Fig. 3.6(a) contiene due MOSFET a canale p, nei quali il transistore TR1 agisce come interruttore elettronico. Il transistore TR2 si comporta da resistenza per il fatto che la tensione del gate è costante. Questo circuito generalmente opera con una notazione logica negativa.

Il circuito della Fig. 3.6(b) è uno sviluppo interessante del primo, dato che contiene sia un dispositivo a canale n sia un dispositivo a canale p. Entrambi i transistori sono comandati da un segnale d'ingresso *A*, e quando uno dei transistori è acceso, l'altro è spento. Quando il segnale di ingresso è lo "0" logico, il transistore TR3 è spento e il transistore TR4 è acceso; in questo stato di funzionamento la linea d'uscita è collegata al polo positivo dell'alimentazione ("1" logico). Quando il segnale
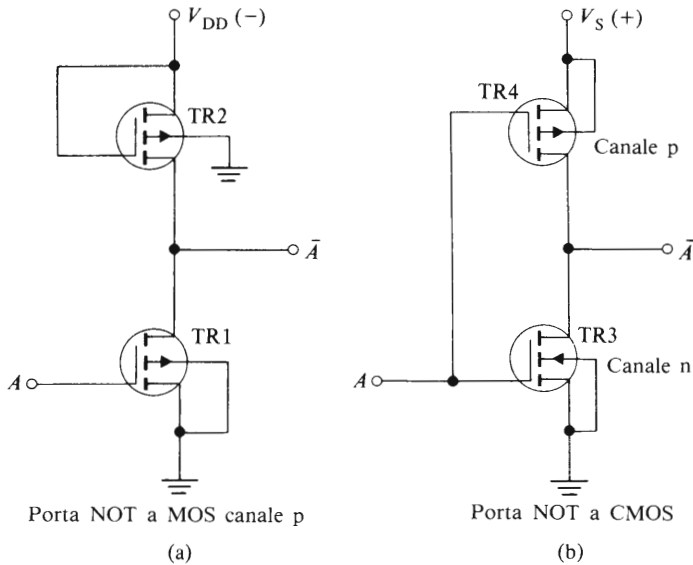
Fig. 3.6 Porte NOT a MOS

d'ingresso è l'"1" logico, il transistore TR3 è acceso e il transistore TR4 è spento; in questo caso la linea di uscita è collegata con la terra (lo "0" logico).

Dato che un MOSFET viene messo in funzione dal segnale di gate, lo descriveremo come un dispositivo attivo. Quando un transistore è utilizzato per sostituire un resistore come carico (è il caso del transistore TR4 della Fig. 3.6), ci riferiamo al transistore come a un carico attivo.

Il circuito della Fig. 3.6(b) è noto come una porta NOT a MOS complementari (CMOS) dato che contiene entrambi i MOSFET a canale p ed a canale n (descritti come una coppia complementare di MOSFET). Altri circuiti CMOS saranno descritti nel capitolo 4.

3.8 La conversione di un circuito a relè in forma logica

Molti circuiti industriali sono controllati da relè; studiamo ora come un circuito con contatti ad autoritenuta (Fig. 3.7a) viene convertito in forma logica.

Sia il circuito a relè che il circuito logico hanno bisogno di un pulsante di arresto e di un pulsante di marcia; la versione a relè nella Fig. 3.7(a) richiede inoltre un insieme di contatti di ritenuta (C) che vengono

chiusi quando la bobina è alimentata.

Nei circuiti logici si assume generalmente che, nello stato di operazione normale, i contatti siano "aperti" e che l'uscita da essi sia lo "0" logico. Tuttavia, nel caso del pulsante di arresto, i contatti sono normalmente chiusi, così che l'uscita dall'interruttore di arresto è l'"1" logico. Quindi designeremo l'interruttore di arresto come $\overline{S1}$ (NOT $S1$) per indicare che è chiuso normalmente e che l'uscita normale da esso è l'"1" logico.

Nel circuito a relè della Fig. 3.7(a) vediamo che la bobina del relè è alimentata *se l'interruttore di ARRESTO è nella posizione normale* **(AND)** *e l'interruttore di MARCIA è stato premuto* ($S2$ = 1) *oppure* **(OR)** *il conduttore è stato alimentato* (C = 1). Cioè

$$C = \overline{S1} \text{ AND } (S2 \text{ OR } C) = \overline{S1}.\ (S2 + C)$$

(a)

(b)
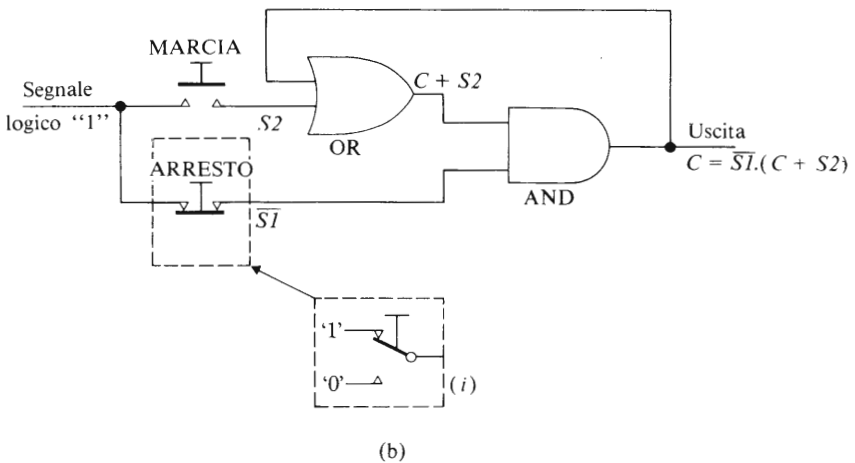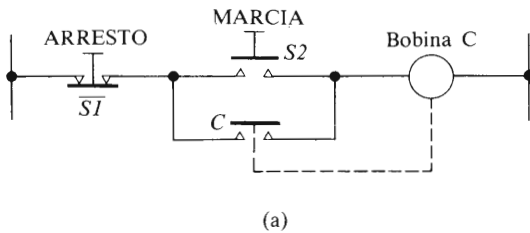
Fig. 3.7 Il circuito a relè in (a) è trasformato nel circuito logico in (b)

Per generare la funzione logica ($S2+C$) abbiamo bisogno di una porta OR a due ingressi con le linee di ingresso alimentate da $S2$ e da C (Fig. 3.7b), dove il segnale C ha lo stesso valore dell'uscita dal sistema. Infatti il segnale C assolve nel circuito logico la stessa funzione assolta dal contatto di ritenuta C nel circuito a relè.

Per completare il circuito logico, l'uscita della porta OR è applicata a una porta AND insieme con il segnale dell'interruttore di ARRESTO $\overline{S1}$. Il segnale C della porta AND è sotto forma di una tensione elettrica che può essere utilizzata per far funzionare una bobina di relè o, in alternativa, per accendere un dispositivo elettronico di potenza, come un tiristore o un triac, che controlli direttamente la corrente principale.

In pratica è necessario assicurare che a ciascun ingresso di ogni porta sia applicato un segnale logico "1" o un segnale logico "0". Se una linea d'ingresso è aperta, la porta funzionerà come se questa linea non esistesse.

Consideriamo, per esempio, la porta logica AND a diodo-resistore della Fig. 3.2; se una linea di ingresso A fosse aperta la porta funzionerebbe semplicemente con il solo segnale B come segnale d'ingresso. L'uscita della porta sarebbe allora $G=B$. Per questa ragione è importante assicurare che ciascuna linea operi con un segnale logico ad essa applicato; un circuito aperto può portare a un cattivo funzionamento del sistema.

Quando la precedente condizione porta a un circuito il cui comportamento non sia sicuro, il sistema deve essere progettato nuovamente. Per esempio, il sistema della Fig. 3.7(b) non è un sistema sicuro: infatti i lettori ricorderanno che, per avere un'uscita da una porta AND corrispondente allo "0" logico, è necessario che uno "0" logico venga applicato ad almeno uno degli ingressi. Nel caso considerato, se premiamo il bottone di ARRESTO apriamo semplicemente una delle linee d'ingresso della porta AND e non applichiamo uno "0" logico ad essa. Di conseguenza l'uscita C rimane all'"1" logico e il teleruttore rimane carico.

Questo problema può essere superato sostituendo l'interruttore di ARRESTO con un interruttore unipolare a due posizioni mostrato nella parte (*i*) della Fig. 3.7(b).

E' buona norma separare i circuiti di sicurezza, come quelli di ARRESTO e di ARRESTO D'EMERGENZA, dai circuiti di marcia, e assicurare che essi stacchino direttamente l'alimentazione dai teleruttori.

3.9 Ulteriori proprietà richieste per un sistema logico

Molte delle famiglie logiche descritte in questo libro richiedono un'alimentazione stabilizzata la cui tensione d'uscita non vari più del 5% circa dal valore nominale. In questi casi è preferibile alimentare i dispositivi di uscita, come relè, elettrovalvole, riscaldatori, per mezzo di alimentazioni separate non stabilizzate. Inoltre, si dovrà usare un comune punto di terra; se ciascuna alimentazione ha un punto di terra, nelle "maglie di terra" risultanti circolano correnti che possono generare problemi.

Segnali di interferenza elettrica possono entrare in un sistema logico da un grande numero di sorgenti. I principali effetti di questi segnali possono essere ridotti al minimo con i seguenti accorgimenti:

1. Chiudere il sistema logico in un contenitore di acciaio collegato a terra.
2. Collocare in un quadro separato relè, teleruttori e dispositivi in alternata.
3. Posare i cavi per l'alimentazione in continua dei sistemi logici in condotti d'acciaio posti a terra.
4. Impiegare cavi schermati per collegare i sensori al sistema logico.
5. Montare i circuiti di filtro il più vicino possibile al sistema logico.
6. Ridurre al minimo i disturbi dovuti all'alimentazione di potenza disponendo opportuni circuiti di filtro tra le linee di alimentazione in alternata.

I circuiti integrati digitali possono facilmente venir danneggiati da un'eccessiva differenza di potenziale e bisogna prendere precauzioni perché ciò non accada.

Interrompere la corrente nei conduttori e nelle bobine del relè può produrre alte tensioni, perciò in parallelo ad essi dovrebbero venire posti un diodo o un condensatore. Alte tensioni possono venire create inoltre dall'uso di apparecchiature di test e di saldatori difettosi; le loro parti metalliche dovranno quindi essere collegate a terra.

Capitolo 4

Porte logiche NAND e NOR

4.1 La ragione dell'esistenza di porte NAND e NOR

I sistemi descritti fin qui sfruttano tre tipi di porte logiche: AND, OR e NOT. Per occuparsi di una vasta gamma di applicazioni dei circuiti logici, è necessario avere a disposizione una quantità adeguata di tutti tre i tipi di porte. Sarebbe vantaggioso che questi tre tipi di porte potessero venire sostituiti dalla combinazione di un insieme di singole porte universali. Queste porte universali esistono, sono note come NAND e NOR e costituiscono l'argomento di questo capitolo. Le porte NAND e NOR sono spesso usate in forma di circuiti integrati.

4.2 Il principio della porta NAND

Collegando una porta AND e una porta NOT come nella Fig. 4.1(a), l'uscita dalla combinazione è la funzione NOT applicata all'uscita della porta AND. La rete risultante è una porta NAND, i cui simboli tipici sono mostrati in Fig. 4.1(b) e (c). In pratica la porta è un singolo circui-
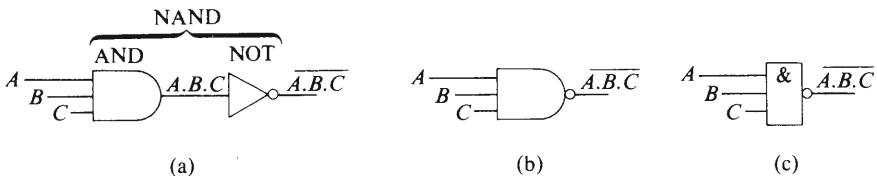
Fig. 4.1 La porta NAND

to unificato, e non si presenta nella forma di due porte separate come qui schematizzato. In questo capitolo si descrivono i circuiti di alcune tipiche porte logiche.

La tabella della verità per una porta logica NAND a tre ingressi è mostrata nella Tabella 4.1. L'uscita, F, viene derivata da

$$F = \text{NOT } (A \quad \text{AND} \quad B \quad \text{AND} \quad C) = \overline{A.B.C}$$

La tabella della verità può venire riassunta come segue. **L'uscita da una porta NAND è un "1" logico se almeno una delle linee d'ingresso ha uno "0" logico applicato ad essa. L'uscita è lo "0" logico soltanto quando tutte le linee d'ingresso sono l'"1" logico.**

Tabella 4.1 Tabella della verità di una porta NAND a 3 ingressi

Ingressi			*Funzione* AND *intermedia*	*Uscita della porta* NAND $F = \overline{A.B.C}$
A	B	C		
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0

4.3 Circuiti tipici contenenti la porta NAND

La Fig. 4.2(b) mostra un tipo di porta NAND a logica positiva realizzata con tecnologia **logica a diodi e transistori** (DTL). La funzione AND è generata dai due diodi collegati agli ingressi A e B, insieme con la catena di resistori che include la resistenza R. La funzione NOT è realizzata dal transistore insieme con il carico del suo collettore.

Entrambi i circuiti della Fig. 4.2 sono costituiti di singoli diodi, resistori e transistori e vengono descritti come *circuiti a componenti discreti*; sono tipici di molte delle prime versioni delle porte logiche. Entrambi i circuiti richiedono un'alimentazione principale (V_{CC}) e un'alimentazione ausiliaria ($-V_{BB}$); quest'ultima viene fornita sia per migliorare la velocità di accensione e spegnimento del circuito, sia per migliorare l'immunità al rumore del circuito stesso.

Le versioni integrate della porta logica DTL richiedono soltanto un
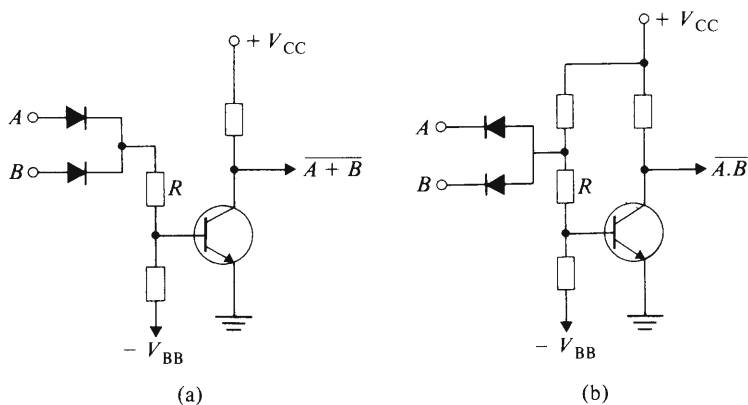
Fig. 4.2 Porte logiche a diodi e transistori (DTL): (a) porta NOR, (b) porta NAND

generatore di potenza e sono assai usate; hanno infatti soppiantato i circuiti a componenti discreti. Sono, come la maggior parte dei dispositivi integrati, caratterizzate dalle dimensioni fisiche molto ridotte e da una velocità d'accensione e spegnimento molto elevata.

La Fig. 4.3 mostra uno dei tipi più comuni di porta logica, nota come
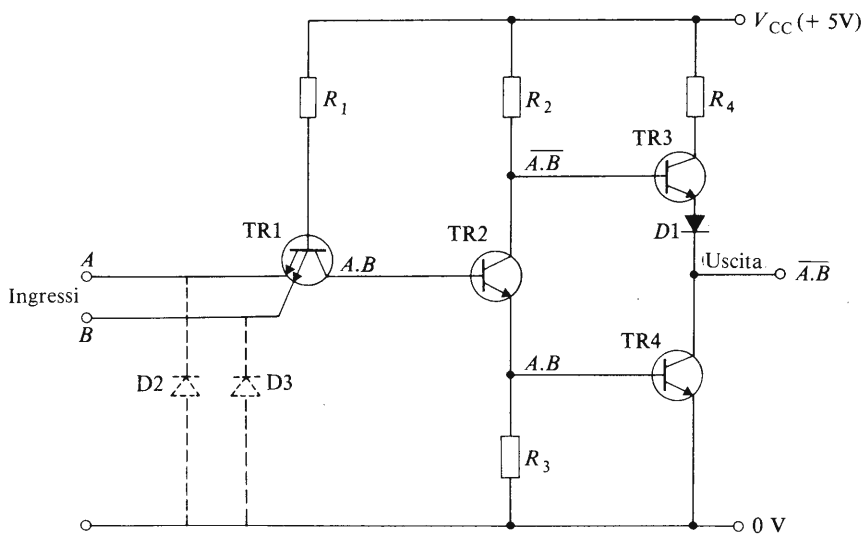
Fig. 4.3 Una porta NAND TTL

porta NAND a **logica transistore-transistore** (TTL). La funzione AND degli ingressi A e B è generata dal collettore di un *transistore TR1 a emettitore multiplo*. Il segnale AND ($A.B$) è applicato alla base del transistore TR2; in base al modo con cui questo transistore è collegato, la funzione AND appare all'emettitore di R2 e la funzione NAND $(\overline{A.B})$ al suo collettore.

La porta NAND TTL della Fig. 4.3 ha un carico di collettore attivo nella forma del transistore TR3; la resistenza R_4 ha un valore molto basso (di solito 130 Ω) e viene aggiunta per limitare la corrente, mentre il diodo D1 viene aggiunto per fornire un certo grado di immunità al rumore.

Quando le condizioni d'ingresso danno $A.B = 1$, il segnale $A.B$ nella base di TR4 è l'"1" logico, il che provoca la sua accensione; questa operazione mette in collegamento la linea di uscita con la linea dello zero volt (quindi l'uscita è uno "0" logico). Nello stesso tempo il segnale $\overline{A.B}$ nella base di TR3 è lo "0" logico, cosicché TR3 viene spento e si comporta come un circuito aperto. In realtà TR3 e TR4 possono essere considerati come interruttori sincronizzati comandati da TR2, nel senso che TR3 è aperto quando TR4 è chiuso e viceversa (Fig. 4.4). Questo circuito di uscita abbastanza insolito viene talvolta descritto come "circuito di uscita totem pole".
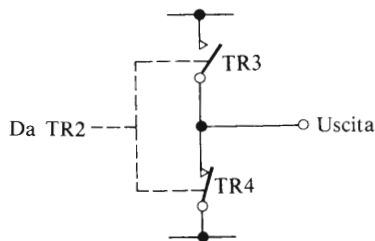
Fig. 4.4 Equivalente elettrico del circuito di uscita della Fig. 4.3

Le porte logiche TTL inoltre incorporano i diodi D2 e D3 per la protezione degli ingressi per la seguente ragione. Sebbene la porta operi con un generatore di tensione positivo, è possibile che in certe condizioni di funzionamento appaia alle linee d'ingresso, per un breve periodo, una tensione negativa. La funzione dei diodi D2 e D3 è quella di proteggere il transistore TR1 dai danni che possono derivare da questi impulsi di tensione negativa.

Le linee d'ingresso inutilizzate sulle porte logiche NAND TTL posso-

no essere trattate nello stesso modo in cui venivano trattate quelle per le porte AND nel paragrafo 3.5. Cioè: (1) possono essere collegate al morsetto positivo del generatore; (2) possono essere collegate in parallelo con una linea d'ingresso utilizzata (si ricordi che ciascuna linea d'ingresso aggiunge un'unità di carico alla porta pilotante). Ogni ingresso di una porta completamente inutilizzata in un circuito integrato TTL può essere lasciato fluttuante o può venir collegato alla linea di zero volt.

Un'altra porta NAND con un carico di collettore attivo è la porta NAND a **MOS complementare** (CMOS) della Fig. 4.5(b). Questa porta usa tipi complementari di MOSFET (cioè dispositivi a canale n e a canale p) nei quali i dispositivi a canale p possono essere considerati cari-
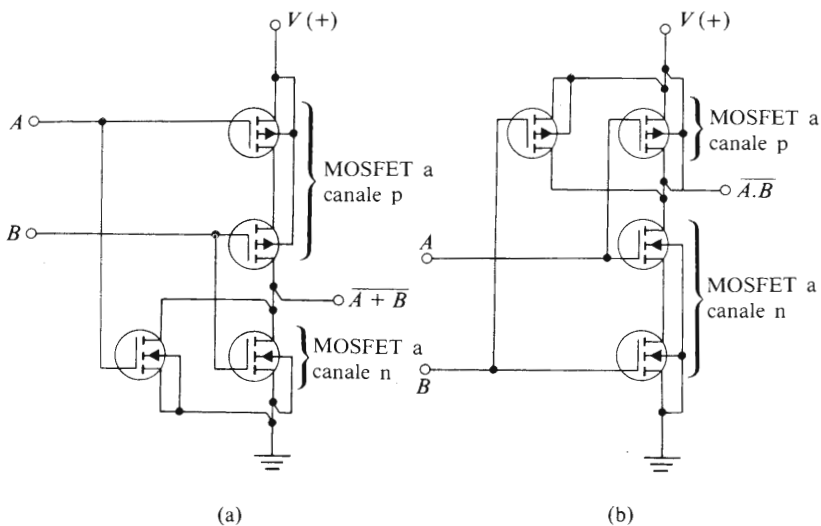
Fig. 4.5(a) Porta NOR a CMOS; (b) Porta NAND a CMOS

chi attivi per i dispositivi a canale n. Un vantaggio di questa famiglia di porte logiche è che, siccome la porta è isolata dal canale conduttore, la corrente che fluisce in ciascuna linea d'ingresso è praticamente nulla. Come risultato la potenza richiesta per far funzionare una linea d'ingresso è molto bassa, consentendo un'elevata capacità di fan-out. Inoltre, dato che i dispositivi a canale p sono spenti quando i dispositivi a canale n sono accesi, la corrente assorbita dal generatore è anch'essa molto bassa. Pertanto si ha un consumo di potenza molto basso, il che rende questi circuiti particolarmente adatti per apparecchi che lavorano a batteria, quali orologi da polso e calcolatori da tavolo.

4.4 Il principio della porta NOR

Combinando una porta OR e una porta NOT come in Fig. 4.6(a), l'uscita dalla combinazione è la funzione NOT dell'uscita dalla porta OR. La rete che risulta è nota come porta NOR, i cui simboli tipici sono mostrati nella Fig. 4.6(b) e (c). Come la porta NAND, in pratica la porta NOR è un singolo circuito unificato; ne riportiamo un esempio nella Fig. 4.5(a).
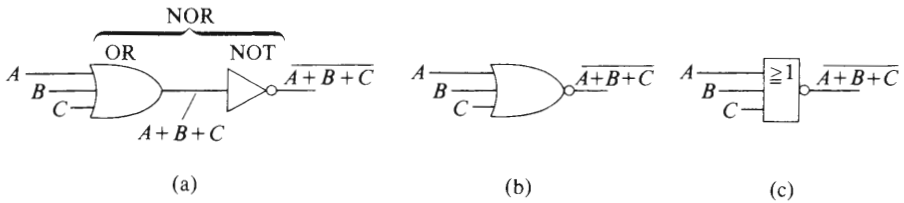
Fig. 4.6 La porta NOR

La tabella della verità per una porta NOR a tre ingressi è riportata nella Tabella 4.2. Gli stati logici della colonna dell'uscita sono derivati dal fatto che

$$\text{Uscita } X = \text{NOT } (A \text{ OR } B \text{ OR } C) = \overline{A+B+C}$$

La tabella della verità della porta NOR può venire riassunta come segue. **L'uscita è lo "0" logico se almeno una delle linee d'ingresso ha un "1" logico applicato ad essa. L'uscita è un "1" logico soltanto quando tutti gli ingressi sono allo "0" logico.**

Tabella 4.2 Tabella della verità di una porta NOR a 3 ingressi

Ingressi			*Funzione* OR *intermedia*	*Uscita*
A	B	C		$X = \overline{A+B+C}$
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

4.5 Circuiti tipici contenenti la porta NOR

Un tipo di porta logica NOR a diodo-transistore a componenti discreti è quello della Fig. 3.2(a). I diodi d'ingresso generano la funzione OR, mentre il transistore fornisce la funzione NOT. Una versione integrata a MOS complementare è invece quella della Fig. 4.5(a).

4.6 Utilizzo delle porte NAND

Considereremo ora come possiamo usare combinazioni di porte NAND per sostituire porte NOT, AND e OR.

La funzione NOT

Consideriamo la porta NAND della Fig. 4.5(a) nella quale soltanto l'ingresso (A) è applicato alla porta. La funzione generata dalla porta può venire dedotta dalla tabella della verità (Tabella 4.1); quando almeno una linea d'ingresso (la sola linea d'ingresso in questo caso) è lo "0" logico, allora l'uscita è l'"1" logico, e quando tutte le linee d'ingresso (l'unica linea d'ingresso in questo caso) sono l'"1" logico, l'uscita è lo "0" logico. Dato che c'è una sola linea d'ingresso, il segnale d'uscita è sempre la funzione NOT dell'ingresso. Quindi una porta NAND con un sigolo ingresso funziona come una porta NOT.

E' buona norma collegare le linee d'ingresso inutilizzate a un segnale logico "1" (connessioni tratteggiate della Fig. 4.7a) o alla linea d'ingresso utilizzata.

La funzione AND

Consideriamo il circuito della Fig. 4.7(b). La porta G1 funziona come una porta NAND, fornendo un'uscita $\overline{A.B}$. La porta G2 ha una sola linea d'ingresso, e perciò funziona come una porta NOT il cui segnale d'uscita è NOT $(\overline{A.B}) = \overline{\overline{A.B}}$. Dalla tabella della verità (Tabella 4.3) si può vedere che l'uscita della porta G2 corrisponde alla tabella della verità della porta A AND B. Perciò una porta AND a due ingressi può essere sostituita da una combinazione di porte NAND come nella Fig. 4.7(b).

Se devono essere ricevuti più segnali d'ingresso, la porta G1 deve avere ulteriori linee di ingresso collegate ad essa. Perciò una porta NAND a quattro ingressi deve essere impiegata se sono quattro i sensori d'ingresso applicati ad essa (connessioni tratteggiate nella Fig. 4.7b).
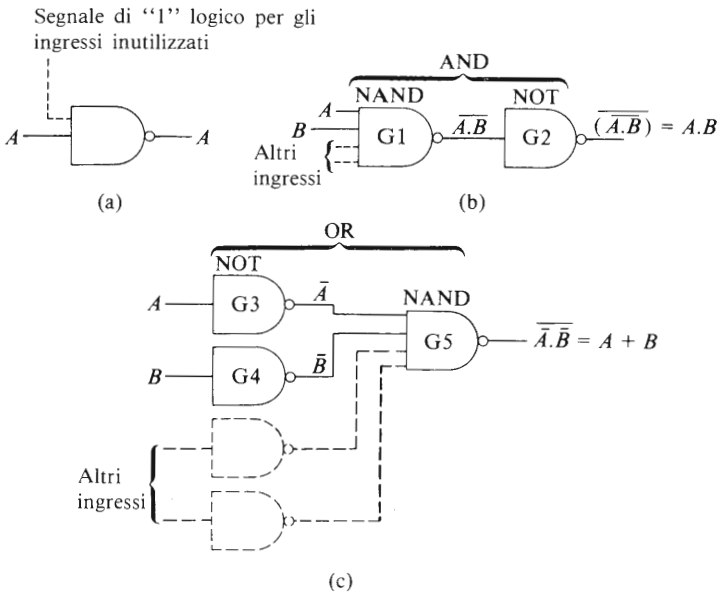
Fig. 4.7 Metodi per generare (a) la funzione NOT, (b) la funzione AND, (c) la funzione OR, utilizzando solo porte NAND

Tabella 4.3 Tabella della verità del circuito della Fig. 4.8 (b)

Ingressi		*Uscita da G1*	*Uscita da G2*
A	*B*	$=\overline{A.B}$	$=\text{NOT } (\overline{A.B}) = \overline{\overline{A.B}} = A.B$
0	0	1	0
0	1	1	0
1	0	1	0
1	1	0	1

La funzione OR

Nella Fig. 4.7(c), le porte G3 e G4 funzionano come porte NOT, cosicché i segnali d'ingresso della porta G5 sono $\bar{A}$ e $\bar{B}$. La tabella della verità per la Fig. 4.7(c) è riportata nella Tabella 4.4. Il lettore faccia attenzione alla colonna degli ingressi A e B e alla colonna di uscita; noterà

Tabella 4.4 Tabella della verità del circuito della Fig. 4.7 (c)

Ingressi		*Uscita da G3 e G4*		*Funzione*	*Uscita da G5*
A	B	$\overline{A}$	$\overline{B}$	$\overline{A}.\overline{B}$	$=$NOT $(\overline{A}.\overline{B}) = A+B$
0	0	1	1	1	0
0	1	1	0	0	1
1	0	0	1	0	1
1	1	0	0	0	1

che la tabella della verità corrisponde a quella di una porta OR a due ingressi. Per questo motivo una porta OR a due ingressi può essere sostituita da tre porte NAND collegate come nella Fig. 4.7(c).

Se devono essere ricevuti più segnali di ingresso, ciascun ingresso addizionale deve venir collegato alla porta G5 attraverso una porta NAND (che funziona da porta NOT). Perciò una porta OR a quattro ingressi viene riprodotta utilizzando quattro porte NAND invertenti, ciascuna avente un singolo ingresso da un sensore; le uscite dagli invertitori sono connesse alla porta G5 (connessioni tratteggiate nella Fig. 4.7c).

In un sistema logico reale vengono utilizzati circuiti integrati, ciascuno dei quali contiene più porte. Un circuito integrato comune è l'SN7400, che contiene quattro porte NAND a due ingressi ed è perciò detto, con terminologia inglese, Quad 2-input NAND (Fig. 4.8). L'incavo a un'estremità dell'integrato permette una corretta collocazione dell'inte-
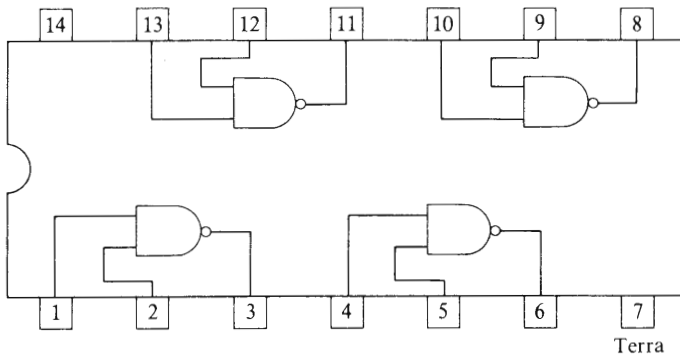
Fig. 4.8 L'integrato SN7400 - NAND con quattro porte a due ingressi in contenitore a 14 piedini in doppia fila

grato nella sede predisposta sul circuito stampato. Conviene notare che ogni qualvolta una porta TTL commuta da uno stato a un altro, si ha una variazione di corrente nella linea che fornisce la tensione. Per superare i problemi associati con queste variazioni di corrente, è consigliabile "filtrare" la linea che fornisce la tensione, collegando tra l'alimentazione e la terra un condensatore di circa 0,01 ÷ 0,1 μF ogni cinque circuiti integrati o anche meno.

4.7 Utilizzo delle porte NOR

In questo paragrafo considereremo le combinazioni di porte NOR che possono essere utilizzate per sostituire porte AND, OR e NOT.

La funzione NOT

Consideriamo la porta NOR della Fig. 4.9(a), in cui è utilizzata una sola linea d'ingresso. Il lettore si ricorderà che per una porta NOT l'uscita è lo "0" logico quando almeno una linea d'ingresso è l'"1" logico e che l'uscita è l'"1" logico quando tutte le linee d'ingresso sono lo
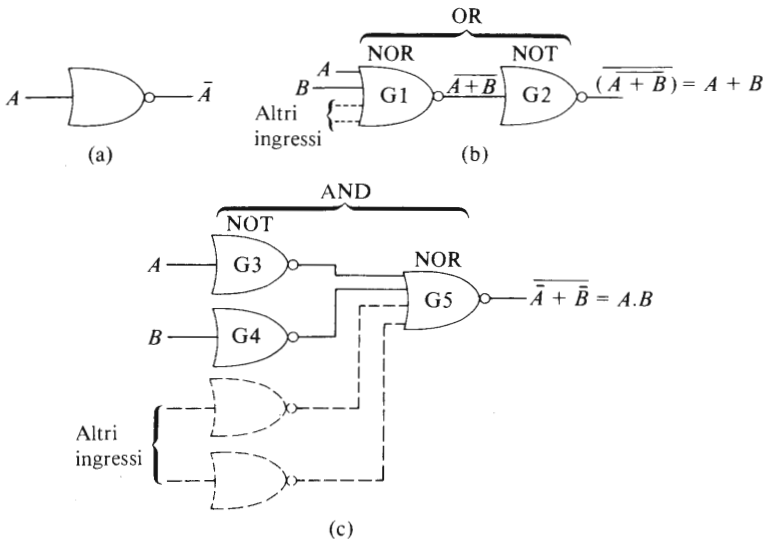
Fig. 4.9 Metodi per generare (a) la funzione NOT, (b) la funzione OR, (c) la funzione AND, utilizzando solo porte logiche NOR.

"0" logico. Dato che la porta considerata ha solo una linea d'ingresso, si comporta come una porta NOT.

La funzione OR

Consideriamo il circuito della Fig. 4.9(b), nel quale G1 si comporta da porta NOR e G2 da porta NOT. La tabella della verità per il circuito è riportata nella Tabella 4.5. Il lettore osserverà che l'uscita della porta G2 (= NOT $(\overline{A+B})$) corrisponde alla funzione OR degli ingressi A e B. Perciò una porta OR a due ingressi può essere sostituita dalla combinazione di porte NOR indicata nella Fig. 4.9(b).

Tabella 4.5 Tabella della verità del circuito della Fig. 4.9 (b)

Ingressi		*Uscita da G1*	*Uscita da G2*
A	B	$= A + B$	$= \text{NOT}(\overline{A+B}) = \overline{\overline{A+B}} = A+B$
0	0	1	0
0	1	0	1
1	0	0	1
1	1	0	1

Se devono essere ricevuti ulteriori segnali d'ingresso, essi possono essere collegati ad altre linee d'ingresso di G1 (linee tratteggiate della Fig. 4.9b). Se si utilizzano quattro linee d'ingresso separate, l'uscita è $A+B+C+D$.

La funzione AND

La funzione AND a due ingressi è generata dalla rete NOR della Fig. 4.9(c); la tabella della verità del circuito è riportata nella Tabella 4.6. Le porte G3 e G4 si comportano da porte NOT, e le uscite da queste porte ($\overline{A}$ e $\overline{B}$ rispettivamente) vengono applicate agli ingressi di G5. Quest'ultima porta genera la funzione logica NOT ($\overline{A}$ OR $\overline{B}$), che il lettore riconoscerà dalla Tabella 4.6 essere la funzione AND degli ingressi A e B.

Se devono essere ricevuti altri segnali d'ingresso, ciascun ingresso addizionale è collegato con una porta NOR (che agisce da porta NOT) e l'uscita di questa porta è collegata a un ingresso di G5 (connessioni tratteggiate della Fig. 4.9c). Se colleghiamo quattro segnali d'ingresso A, B, C, D, l'uscita della porta G5 è $A.B.C.D$.

Tabella 4.6 Tabella della verità del circuito della Fig. 4.9 (c)

Ingressi		*Uscita da G3 e G4*		*Funzione*	*Uscita da G5*
A	B	$\bar{A}$	$\bar{B}$	$\bar{A}+\bar{B}$	$=$ NOT $(\bar{A}+\bar{B}) = A.B$
0	0	1	1	1	0
0	1	1	0	1	0
1	0	0	1	1	0
1	1	0	0	0	1

4.8 Un semplice metodo di minimizzazione

Ci si può chiedere perché si usino tre porte NOR per sostituire una porta AND, quando sarebbe più semplice utilizzare una sola porta AND. Molto semplicemente, la risposta è che generalmente è possibile ridurre o "minimizzare" il numero di porte di una rete complessa, cosicché abbiamo bisogno soltanto di tante porte NOR (o di porte NAND) quante porte AND, OR e NOT messe assieme. E' abbastanza strano, ma certe tecniche ci permettono talvolta di utilizzare meno porte NOR (o NAND) di quante ne avremmo utilizzate usando le AND, OR e NOT! In questo paragrafo considereremo soltanto una tecnica di base molto semplice che permette di illustrare il principio generale.

Consideriamo i due circuiti della Fig. 4.10, ciascuno dei quali ha due porte NOT in cascata. Due inversioni logiche avvengono tra l'ingresso e
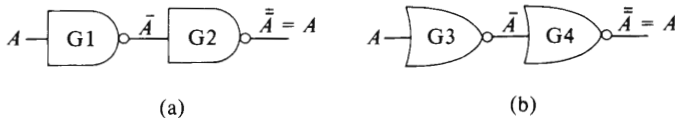
Fig. 4.10 Una semplice tecnica di minimizzazione mediante (a) porte NAND, (b) porte NOR.

l'uscita, cosicché uno "0" logico in A appare come un "1" logico in $\bar{A}$ e riappare come uno "0" logico all'uscita. Quindi

$$\text{NOT (NOT } A) = \bar{\bar{A}} = A$$

Analogamente, un "1" logico in A appare come un "1" logico all'uscita. Chiaramente le porte G1 e G2 possono venire eliminate e so-

stituite con un conduttore semplice, come le porte G3 e G4. In entrambi i casi abbiamo potuto eliminare due porte logiche dal circuito. Un'applicazione di questa tecnica è illustrata nel paragrafo 4.12.

4.9 Un oscillatore a onde quadre

Si può costruire un semplice oscillatore a onde quadre usando due porte NOR e alcuni altri componenti, come mostrato nella Fig. 4.11. Un circuito simile può essere facilmente costruito utilizzando porte NAND.
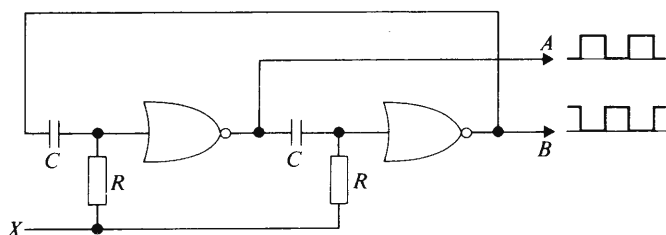
Fig. 4.11 Un oscillatore a onde quadre o generatore di impulsi di "clock"

Le porte NOR sono collegate da circuiti contenenti resistenze e condensatori (RC) che controllano la sequenza temporale delle onde quadre generate dal circuito. I segnali di uscita *A* e *B* sono segnali logici complementari, cosicché quando l'uscita in *A* è l'"1" logico, l'uscita in *B* è lo "0" logico e viceversa. L'uscita da *A* oppure da *B* può essere utilizzata come sorgente di segnali di *clock* per alcuni dei circuiti che verranno descritti più avanti. Il circuito mostrato nella Fig. 4.1 è un tipo di **multivibratore astabile.**

Il circuito funziona come un oscillatore quando un "1" logico viene applicato alla linea *X*. Quando il segnale applicato a questa linea è lo "0" logico, l'oscillatore smette di funzionare. In questo modo è possibile utilizzare il segnale applicato alla linea *X* per controllare il funzionamento del generatore di segnali di clock.

4.10 Introduzione al progetto di reti logiche (sintesi di reti)

Un problema logico può venir presentato in forma di parole scritte o in forma di tabella della verità. Le affermazioni logiche sono solitamente meglio comprese sotto forma di tabella della verità, cosicché quando

il problema appare come affermazione scritta è consigliabile la sua trascrizione in forma di tabella della verità.

Consideriamo il seguente problema.

Un'organizzazione di sorveglianza vuole fornire un servizio di massima sicurezza, e per fare questo il microcalcolatore utilizzato nelle installazioni di sorveglianza viene triplicato. Occorre un progetto che prenda decisioni sulla validità delle uscite dei tre calcolatori; l'uscita da questo sistema di "voto" (decisione) deve essere "1" se l'uscita da almeno due dei tre calcolatori è l'"1" logico, e l'uscita deve essere lo "0" logico se due o più dei tre calcolatori danno un'uscita "0" logico. Il circuito di "voto" assicura che, in caso di malfunzionamento di uno dei tre microcalcolatori, si può fare assegnamento sul segnale logico d'uscita risultante.

Quando progettiamo una rete logica in base a un'affermazione o a una tabella della verità, diciamo che la rete è stata **sintetizzata**.

Un diagramma a blocchi che rappresenta il sistema è quello della Fig. 4.12, e la tabella della verità del sistema di voto è riportata nella Tabella 4.7. Dato che abbiamo tre calcolatori A, B e C, ciascuno dei quali fornisce un segnale d'uscita indipendente, la tabella della verità contiene tutte le possibili combinazioni di queste variabili. Le combinazioni sono scritte secondo lo schema in paragrafo 2.8.

Dobbiamo ora scrivere gli 1 e gli 0 nella colonna d'uscita (F) corrispondenti alle condizioni specificate dal problema. Il problema implica che l'uscita sia l'"1" logico quando due o più ingressi siano l'"1" logico e queste condizioni si verificano nella quarta, sesta, settima e ottava riga (Tabella 4.7). Inoltre, se due o più ingressi sono lo "0" logico, allora l'uscita deve essere lo "0" logico; questo si verifica nella prima, seconda, terza e quinta riga. Il progetto seguente si basa sul fatto che l'uscita è un "1" se due o più ingressi sono "1". Esprimendoci in parole, diciamo che:

$$F=1 \text{ se } \overline{A}.B.C=1 \text{ OR se } A.\overline{B}.C=1$$
$$\text{OR se } A.B.\overline{C}=1 \text{ OR se } A.B.C=1$$

Quando scriviamo questo in "forma logica", otteniamo

$$F=\overline{A}.B.C+A.\overline{B}.C+A.B.\overline{C}+A.B.C \tag{4.1}$$

La Fig. 4.13 mostra il circuito di voto costruito secondo l'equazione (4.1). In questo circuito la porta G2 genera un'uscita corrispondente ad $\overline{A}.B.C$, la porta G1 fornisce l'ingresso $\overline{A}$ a G2. Il fatto che l'uscita da
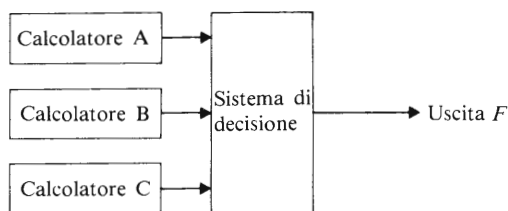
Fig. 4.12 Sistema elettronico di voto

Tabella 4.7 Tabella della verità per un sistema elettronico di "voto"

Ingressi			*Uscita*	*Condizioni d'ingresso che danno l'"1" logico come uscita*
A	*B*	*C*	*F*	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}.B.C$
1	0	0	0	
1	0	1	1	$A.\overline{B}.C$
1	1	0	1	$A.B.\overline{C}$
1	1	1	1	$A.B.C$

questa porta sia l'"1" logico significa che i calcolatori B e C forniscono uscite corrispondenti all'"1" logico, mentre il calcolatore A fornisce un'uscita che è lo "0" logico; questa condizione di funzionamento implica che il calcolatore A sia in errore e perciò l'"1" logico all'uscita di G2 può essere usato per indicare questo errore. Analogamente quando un "1" logico appare all'uscita di G3, questo indica un errore nel calcolatore B. Un "1" logico all'uscita di G4 indica un errore nell'uscita del calcolatore C.

Se tutti i calcolatori funzionano correttamente, l'uscita delle porte G2, G3, G4 deve essere sempre zero. Quando tutti i calcolatori funzionano e ciascuno di essi fornisce un'uscita corrispondente a un "1" logico, la porta G5 fornisce un'uscita che corrisponde a un "1" logico. La porta G6 fornisce la funzione OR delle uscite dalle quattro porte AND.

La rete della Fig. 4.13 perciò non soltanto fornisce un voto sull'uscita dei calcolatori, ma indica anche quale calcolatore dà la risposta sbagliata. Tuttavia, il nostro problema non richiede l'informazione su quale dei
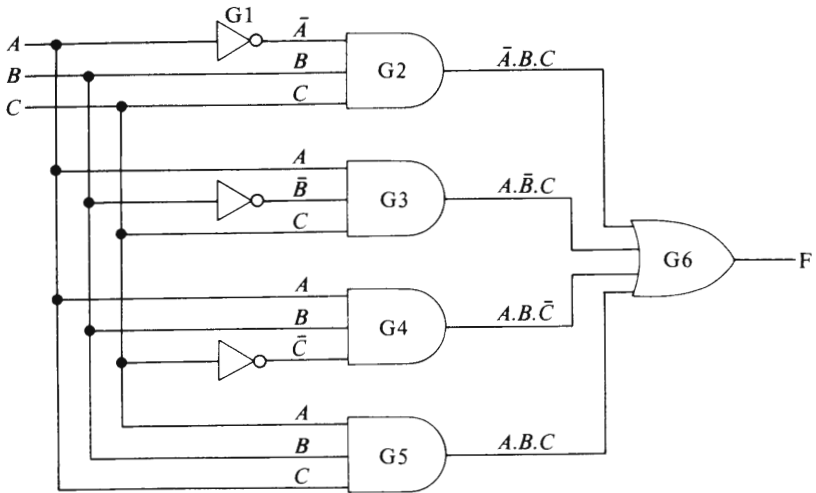
Fig. 4.13 Soluzione circuitale per il sistema elettronico di voto ottenuta direttamente dalla tabella della verità 4.7

calcolatori sia in errore. Perciò dobbiamo fare un'analisi del circuito per vedere se possiamo semplificarlo (minimizzarlo), risolvendo ugualmente il problema.

Un sistema logico che è progettato per un problema specifico, come nel caso appena considerato, è costituito di elementi elettronici di **hardware**, consistenti di porte logiche singole montate su un *circuito stampato* (*p.c.b.*, dall'inglese **printed circuit board**): questo è una piastra costituita di materiale isolante con conduttori metallici (solitamente di rame) ''stampati'' sulla sua superficie tramite un processo fotografico. I componenti vengono inseriti in fori preparati attraverso la superficie e i conduttori, e ivi vengono saldati. Un esempio di circuito stampato è riportato nella Fig. 4.14. Una rete logica costruita in questo modo per risolvere un problema specifico è generalmente nota come circuito realizzato in hardware **(hardwired)**.

4.11 Minimizzazione del circuito elettronico di voto

E' possibile eliminare tutte le porte NOT della Fig. 4.13 ragionando come segue: consideriamo il funzionamento della porta G2. Se la sua uscita è l'''1'' logico, $A = 0$ ($\bar{A} = 1$), $B = 1$, e $C = 1$; cioè, i calcolatori B e
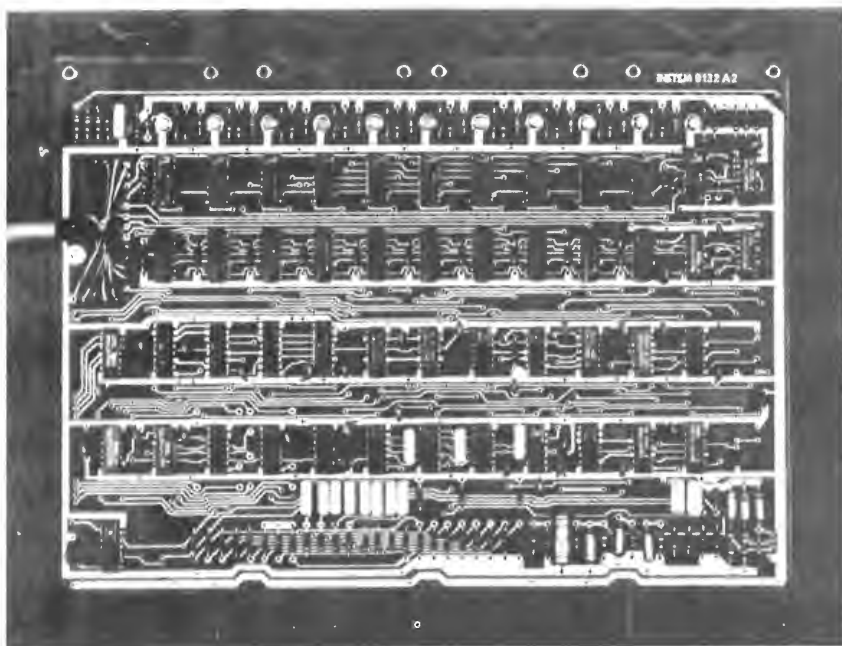
Fig. 4.14 Un circuito stampato contenente circuiti integrati e componenti discreti (riprodotto su gentile concessione della Instem Ltd)

C danno entrambi un "1", mentre il calcolatore A non funziona correttamente e dà un'uscita corrispondente allo "0" logico. Dato che non abbiamo necessità di scoprire un errore nel funzionamento del calcolatore A, possiamo tralasciare l'ingresso $\overline{A}$. Ciò implica che la porta G2 ha bisogno soltanto di due ingressi, B e C, cosicché la sua uscita si riduce a $B.C$ (in definitiva, la sua uscita è "1" quando $B=1$ AND $C=1$).

Analogamente possiamo fare a meno delle restanti porte NOT della Fig. 4.13, cosicché l'uscita da G3 diventa $A.C$, e l'uscita da G4 è $A.B$.

Nella Fig. 4.13, la porta G5 è la sola porta che dà un'uscita "1" logico quando tutti i calcolatori danno l'"1" logico. Tuttavia si può mostrare che questa porta può essere tralasciata. Con le modificazioni menzionate sopra e quando $A=1$, $B=1$, $C=1$, la porta G2 (la cui uscita è ora $B.C$) fornisce un segnale "1", la porta G3 (la cui uscita è ora $A.C$) dà un segnale "1", e la porta G4 (la cui uscita è ora $A.B$) fornisce ancora un segnale "1". Così, usando soltanto le porte G2, G3, G4, modificate in questo modo, se due o più dei calcolatori funzionano correttamente il
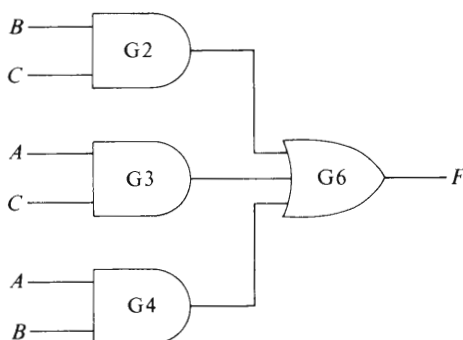
Fig. 4.15 Versione minimizzata del circuito elettronico di voto

segnale corretto appare all'uscita di G6. Il circuito minimizzato è riportato nella Fig. 4.15.

4.12 La trasformazione di una rete AND, OR, NOT in una rete NAND

Una rete contenente porte AND, OR e NOT può essere trasformata in una rete NAND sostituendo ciascuna porta con il suo equivalente NAND (vedi paragrafo 4.6). Dove è possibile si può usare la tecnica di minimizzazione descritta nel paragrafo 4.8.

Applichiamo ora il metodo sopra descritto alla rete di voto della Fig. 4.15. Prima di tutto ridisegniamo il circuito sostituendo a ciascun elemento AND e OR il suo equivalente NAND: pertanto G2 della Fig. 4.14 viene sostituito dalle porte G7 e G7A della Fig. 4.16(a); analogamente G3 della Fig. 4.15 è sostituito dalle porte G8 e G8A, e G4 della Fig. 4.15 è sostituito dalle porte G9 e G9A della Fig. 4.16(a). Inoltre la porta OR G6 della Fig. 4.15 è sostituita da quattro porte NAND nella Fig. 4.16(a).

Il lettore noterà che l'ingresso $\overline{B.C}$ nella porta G7A è logicamente invertito da G7A e nuovamente dalla porta G7B, cosicché l'uscita di G7B è di nuovo $\overline{B.C}$. Chiaramente la coppia di porte G7A e G7B è ridondante e può essere sostituita da un semplice conduttore. Analogamente la coppia di porte G8A e G8B è ridondante, come pure la G9A e G9B. Quando queste porte vengono eliminate, la rete finale appare come nella Fig. 4.16(b).

Nel caso qui considerato, la rete NAND della Fig. 4.16 contiene tante porte logiche quante la rete OR e AND ne conteneva nella Fig. 4.16(b).

(a)

(b)
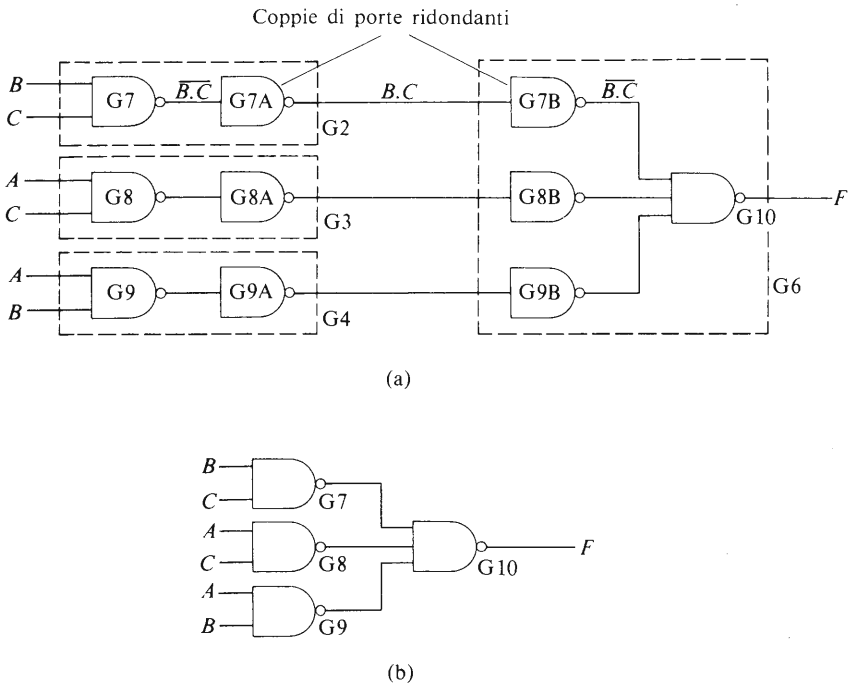
Fig. 4.16 Conversione di un circuito AND e OR in una rete NAND

4.13 La conversione di una rete AND, OR, NOT in una rete NOR

Sostituendo ciascuna porta in una rete AND, OR, NOT con il suo equivalente NOR (vedi paragrafo 4.7), la rete NOR che ne risulta è equivalente alla rete originale.

Per esempio, disegniamo una rete NOR la cui uscita è data dall'espressione logica:

$$F=(A+C).(B+C).(A+B)$$

La rete AND e OR di base che soddisfa questa espressione è riportata nella Fig. 4.17(a), nella quale la porta OR G1 fornisce la funzione $A+C$, G2 fornisce la funzione $B+C$, G3 quella $A+B$. Le uscite da queste porte sono collegate alla porta G4 che funziona da porta AND per dare l'uscita desiderata.
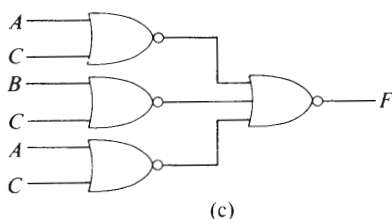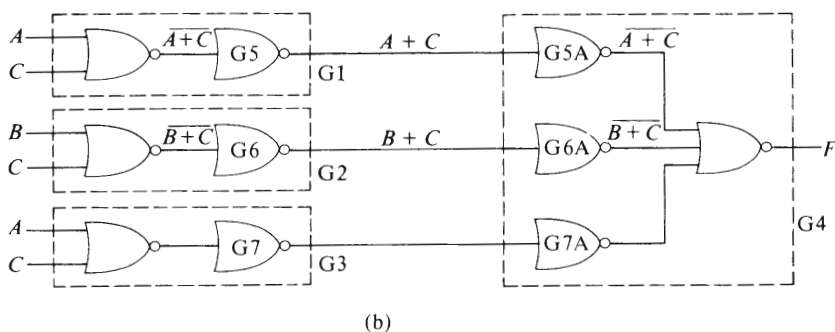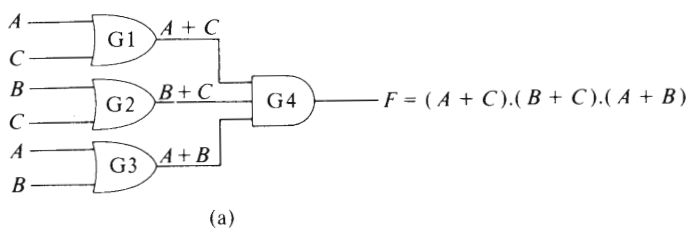
Fig. 4.17 Sostituzione di una rete AND e OR con una rete NOR

Ciascuna porta è sostituita dal suo equivalente NOR nella Fig. 4.17(b) (vedi anche Fig. 4.9). Applicando la tecnica della ridondanza descritta nel paragrafo 4.8, vediamo che le porte G5 e G5A sono ridondanti, come le coppie di porte G6 e G6A, G7 e G7A. Quando queste porte vengono eliminate, la rete finale appare come nella Fig. 4.17(c); il lettore noterà che la rete NOR contiene tante porte logiche quante la rete OR e AND ne conteneva nella Fig. 4.17(a).

Il lettore potrà trovare interessante sviluppare la tabella della verità

per il circuito della Fig. 4.17, dato che lo troverà identico a quello della Tabella 4.7 per il circuito elettronico di voto. Infatti i circuiti della Fig. 4.17(a) e (c) sono semplicemente un'altra versione delle reti delle Fig. 4.15 e 4.16(b).

Capitolo 5

Elementi bistabili

5.1 Che cos'è un elemento bistabile?

Un elemento bistabile è un elemento che può operare in due stati stabili diversi. Un semplice esempio di elemento bistabile è l'interruttore elettrico; questo dispositivo può essere nello stato ON o nello stato OFF. Ci sono, naturalmente, intervalli di tempo in cui i contatti dell'interruttore, nel passare da uno stato all'altro, non sono né ON né OFF, ma questi intervalli di tempo sono molto brevi.

Quando l'interruttore si trova nello stato ON o nello stato OFF, diciamo che è in uno stato operativo **stabile** dato che non lo cambia di sua iniziativa.

I dispositivi bistabili possono essere considerati come semplici forme di **elementi di memoria** o di **elementi di immagazzinamento**, dato che rimangono nello stato corrispondente all'ultima istruzione loro fornita. Nei circuiti logici, l'istruzione è sotto forma di segnale logico (che può essere in forma di impulso di brevissima durata — soltanto alcuni microsecondi o perfino nanosecondi in alcuni casi). Questi elementi sono descritti frequentemente come **flip-flop**.

5.2 Il flip-flop set-reset (*S-R*)

Il flip-flop *S-R* è un dispositivo che ha due terminali d'ingresso e due terminali di uscita. Le linee d'ingresso sono contrassegnate dalle lettere S e R (Fig. 5.1), e le linee d'uscita sono contrassegnate dalle lettere Q e $\overline{Q}$ (NOT Q); cioè, i due segnali d'uscita sono complementari (con un'eccezione che verrà descritta più avanti), cosicché quando $Q=0$, allora $\overline{Q}=1$ e viceversa. Mostreremo più avanti che quando un "1" logico è applicato alla linea d'ingresso S, l'uscita di Q viene posta all'"1" logico (in inglese SET = porre). Quando un "1" logico è applicato all'ingresso R, l'uscita di Q viene nuovamente posta a zero (in inglese RESET = porre nuovamente).
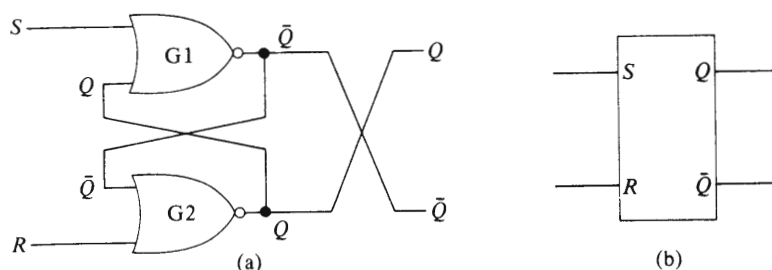
Fig. 5.1 Il flip-flop set-reset (*S-R*): (a) rete logica, (b) simbolo del circuito

Una semplice analogia con il flip-flop *S-R* è rappresentata dal sistema di marcia-arresto di un motore azionato da pulsanti. Possiamo considerare il pulsante di "marcia" come l'ingresso *S* e il pulsante di "arresto" come l'ingresso *R*; il motore verrà collegato all'uscita *Q*. Premendo il pulsante di marcia (il che equivale ad applicare un "1" alla linea *S*), si provoca la partenza del motore (il che equivale a porre (set) l'uscita di *Q* nello stato "1"). Premendo il pulsante di arresto (il che equivale ad applicare un "1" alla linea *R*), si provoca la fermata del motore (il che equivale a riportare (reset) l'uscita di *Q* allo stato "0").

Supponiamo che i segnali d'ingresso siano $S=0$ e $R=0$ nell'istante in cui viene fornita l'alimentazione. Il lettore ricorderà che per una porta NOR l'uscita è l'"1" logico quando tutti gli ingressi sono lo "0". All'accensione, sia Q che $\overline{Q}$ sono momentaneamente "0"; il risultato globale è che le porte G1 e G2 fanno a gara tra di loro per raggiungere l'uscita corrispondente all'"1" logico. Quella delle due porte che raggiunge l'"1" logico per prima, ha "vinto" la gara. Supponiamo che $\overline{Q}$ diventi l'"1" logico per primo; questo segnale viene applicato all'ingresso di G2 attraverso il cammino di reazione, e fa sì che l'uscita di G2 diventi "0" (un "1" logico applicato a un qualsiasi ingresso di una porta NOR dà come uscita uno "0" logico). Così, nel circuito della Fig. 5.1(a), l'utilizzatore non ha nessun controllo su quale dei segnali di uscita diventi "1" logico nell'istante di accensione; questo era uno dei problemi delle prime calcolatrici tascabili: nel momento in cui si accendeva la calcolatrice, appariva sul visualizzatore un numero del tutto casuale.

Una volta che la condizione di "gara" iniziale è finita, le connessioni incrociate tra le uscite e gli ingressi delle porte fanno sì che le uscite siano stabili. Nel caso sopra esposto risulta $Q=0$, $\overline{Q}=1$. Operando in modo da mantenere le condizioni d'ingresso costanti nel tempo, $S=0$ e $R=0$, la condizione di funzionamento qui descritta è nota come uno degli **stati stabili**.

Supponiamo ora di applicare un "1" logico alla linea S di G1 (con $R=0$). Questo segnale provoca come risultato il passaggio dell'uscita di $\overline{Q}$ da "1" a "0"; un "1" all'ingresso di una porta NOR genera in uscita il segnale "0"). Nello stesso tempo, il nuovo segnale di uscita $\overline{Q}$ è applicato all'ingresso di G2, cosicché entrambi gli ingressi di G2 sono "0"; il risultato è che l'uscita di G2 cambia da "0" a "1" e cioè l'uscita Q è uguale a "1". L'effetto globale di applicare un "1" logico alla linea S è quello di ottenere che l'uscita Q venga posta (set) a "1". Per questo motivo la linea S può essere considerata la linea che posiziona Q; quando $S=1$, e $R=0$ allora il flip-flop è nel suo **modo di funzionamento "set"**.

Quando il segnale della linea S è riportato a "0", i cammini incrociati di reazione mantengono le uscite nello stato $Q=1$ e $\overline{Q}=0$. Il segnale applicato alla linea d'ingresso S per provocare la sequenza di avvenimenti appena descritti può essere anche di brevissima durata (per molti flip-flop è dell'ordine di alcuni nanosecondi): basta che sia applicato per un tempo sufficientemente lungo perché la sequenza di commutazioni si propaghi attraverso le due porte logiche. Per questa ragione è possibile usare impulsi molto brevi per modificare lo stato dei flip-flop. L'applicazione ripetuta di un "1" logico alla linea S non ha ulteriori effetti sullo stato delle uscite, dato che non fa altro che provare a porre (set) l'uscita Q nello stato "1", che essa ha già assunto.

In modo analogo si può mostrare che l'applicazione di un "1" logico alla linea R (con $S=0$) provoca il ritorno (reset) dell'uscita Q a "0" (e simultaneamente provoca l'assunzione dello stato "1" da parte di $\overline{Q}$). Questo funzionamento è detto **modo di funzionamento "reset"**. Anche l'impulso di reset può essere di brevissima durata.

C'è, tuttavia, un altro modo di funzionamento che si verifica quando applichiamo *simultaneamente* i segnali $S=1$, $R=1$. Quando ciò avviene, le uscite di G1 e G2 sono *entrambe lo "0" logico* simultaneamente.

Sebbene questa condizione di funzionamento non danneggi il circuito, il suo utilizzo è sconsigliabile per la seguente ragione: se, dopo la condizione in cui $S=1$ e $R=1$, cambiamo entrambi i segnali di ingresso nello "0" logico, si genera di nuovo una "gara" tra le uscite per vedere quella che può raggiungere lo stato di "1" logico per prima. Si perde così ogni informazione che possa indicare quale delle due uscite diventerà un "1" logico prima che essa lo diventi effettivamente. Dato che non possiamo prevedere il risultato finale della "gara", è sconsigliabile utilizzare l'insieme di condizioni d'ingresso $S=1$, $R=1$.

La Fig. 5.2 mostra un flip-flop SR NAND controllato dai segnali "1" alle linee R e S. Le porte G1 e G2 possono essere eliminate se vengono usati segnali corrispondenti allo "0" logico per controllare il flip-flop consistente nelle porte G3 e G4.
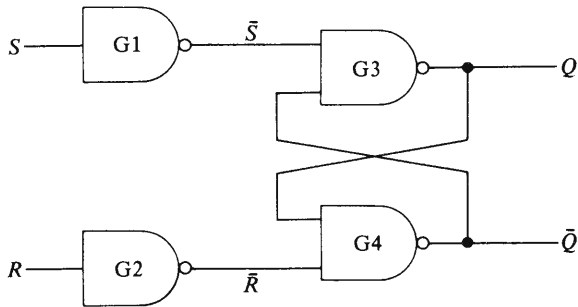
Fig. 5.2 Un flip-flop *S-R* NAND

5.3 Sequenza di controllo di pistoni contenente un flip-flop *S-R*

Consideriamo il funzionamento della sequenza di controllo adatta al controllo di due pistoni A e B. Essi all'inizio devono essere completamente ritratti; dopo che abbiamo premuto il pulsante di marcia, devono completare la sequenza A_+, B_+, B_-, A_-. Le operazioni A_+ e B_+ implicano che i pistoni A e B si muovano uno dopo l'altro fino a raggiungere la loro posizione di massima corsa esterna. Le operazioni B_- e A_- richiedono che i pistoni si ritraggano, in quest'ordine, dalla posizione di massima corsa esterna.

Questa sequenza è tipica di una macchina utensile che (1) blocca un pezzo in lavorazione (operazione A_+), (2) porta una testa di taglio sul pezzo in lavorazione (operazione B_+), (3) ritrae, dopo aver tagliato, la testa di taglio (operazione B_-), (4) libera il pezzo (operazione A_-).

Un diagramma del funzionamento dei pistoni è riportato nella Fig. 5.3, nella quale la posizione di ciascun pistone è rilevata per mezzo di interruttori di fine corsa. Quando il pistone A è completamente ritratto, l'interruttore A_1 fornisce in uscita un "1" logico e A_2 fornisce uno "0" logico. Quando il pistone A è completamente estratto, l'interruttore A_1 fornisce in uscita una "0" logico e A_2 fornisce un "1" logico. Durante il cammino intermedio sia A_1, sia A_2 forniscono un segnale di uscita corrispondente allo "0" logico. Le uscite dagli interruttori B_1 e B_2 sono del tutto analoghe per le posizioni corrispondenti del pistone B. Le uscite dagli interruttori per ciascuno stadio della sequenza operativa sono riportate nella Tabella 5.1.

La fornitura di aria a ciascun pistone è controllata da un'elettrovalvola a due posizioni e quattro vie. Quando si applica un segnale elettrico all'ingresso A_- dell'elettrovalvola SVA, viene immessa aria nel lato de-
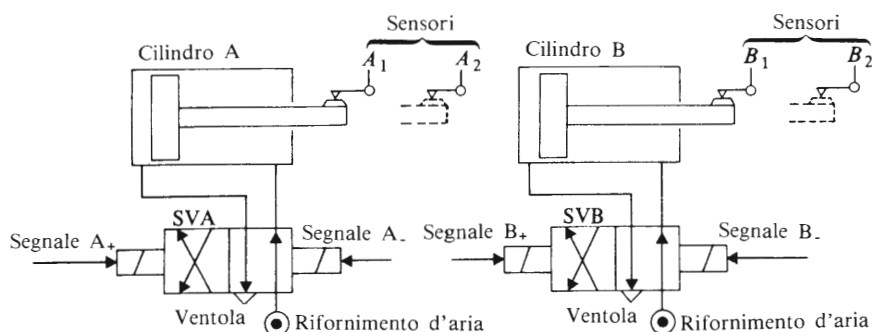
Fig. 5.3 I pistoni e gli interruttori nella loro sequenza di movimento

Tabella 5.1

Stato di funzionamento	*Stato dei sensori* A_1	A_2	B_1	B_2	*Note*
0	1	0	1	0	Sistema fermo
1	0	1	1	0	L'espansione di A è completa; si inizia l'espansione di B.
2	0	1	0	1	L'espansione di B è completa; si inizia la compressione di B.
3	0	1	1	0	La compressione di B è completa; s'inizia la compressione di A.
4	1	0	1	0	La compressione di A è completa; il ciclo è terminato.

stro del pistone A, mentre il lato sinistro si trova a pressione atmosferica. Quando si applica un segnale elettrico all'ingresso A_+ di SVA, si immette aria a sinistra del pistone A e il suo lato destro è a pressione atmosferica.

Il lettore noterà nella Tabella 5.1 che le uscite dagli interruttori sono identiche nei due stati operativi 1 e 3, cioè quando il pistone A è completamente estratto e B è completamente compresso: in tal modo la condizione $A_2 = 1$, $B_1 = 1$ si verifica due volte durante la sequenza completa. Dobbiamo perciò escogitare qualche mezzo per "informare" il sistema logico di quale passo deve intraprendere seguendo questo insieme di con-

dizioni. Se il sistema ha raggiunto lo stato 1, allora deve iniziare l'operazione B_+; se ha raggiunto lo stato 3, deve iniziare l'operazione A_-. Una soluzione di questo problema è riportata nella Fig. 5.4 (si tenga presente che vi sono altre soluzioni possibili).

All'inizio della sequenza, i pistoni sono completamente ritratti ($A_1 = 1$, $B_1 = 1$); l'"1" logico dell'interruttore A_1 mette l'uscita Q del flip-flop *FF* nello stato "1" (cosicché $\bar{Q} = 0$). Premere il pulsante di marcia fa sì che l'uscita da G1 sia un "1" logico; questo segnale viene trasmesso alla bobina A_+ dell'elettrovalvola SVA (Fig. 5.3) e dà inizio all'estrazione del pistone A_+. I blocchi amplificatori di potenza della Fig. 5.4 sono necessari dato che il sistema logico è capace solo di fornire una corrente di pochi milliampere. Per generare una corrente sufficiente a pilotare l'elettrovalvola è necessario amplificare l'uscita di corrente del circuito logico; l'amplificatore di potenza può contenere uno qualsiasi di diversi dispositivi, per esempio un transistore, un tiristore o semplicemente un relè.
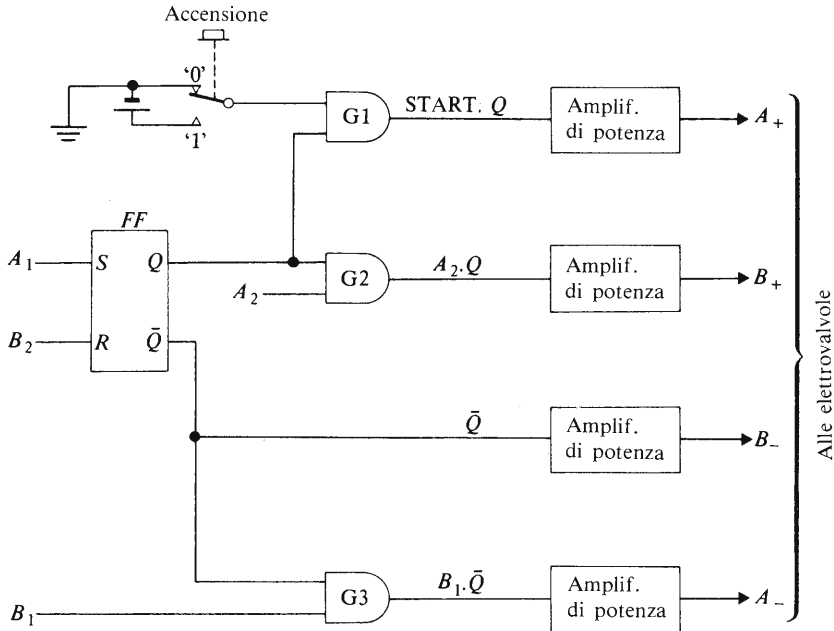
Fig. 5.4 Diagramma a blocchi del sistema logico di controllo della sequenza degli stantuffi

Quando l'estrazione dello stantuffo A_+ è completa, l'uscita dell'interruttore A_2 diventa un "1" logico; questa uscita è inviata insieme con l'uscita Q (dato che il flip-flop non è stato riposizionato, Q rimane "1") alla porta AND G_2 per fornire il segnale che dà inizio alla corsa B_+.

Alla fine della corsa B_+ l'uscita dell'interruttoreB_2 diventa un "1" logico. Questo segnale riposiziona il flip-flop ($Q=0$, $\overline{Q}=1$), con il duplice effetto di inziare la corsa B_-, e di impedire il funzionamento delle porte G1 e G2. In questo modo ora il sistema "conosce" che si è giunti alla seconda parte della sequenza operativa (vedi Tabella 5.1).

Quando la corsa B_- è completa, lo stato degli interruttori è ancora una volta $A_2=1$, $B_1=1$. Tuttavia, dato che $\overline{Q}=1$, l'uscita di G3 diventa l'"1" logico e dà inizio alla corsa A_-; dato che $Q=0$, questa volta non può verificarsi la corsa B_+ (vedi la porta G2). Al completamento della corsa A_-, compressione di A, l'uscita dell'interruttore A_1 diventa l'"1" logico, che posiziona l'uscita Q del flip-flop nello stato "1". Il sistema è perciò pronto per cominciare il ciclo successivo di operazioni, una volta che venga nuovamente premuto il pulsante di marcia.

Il sistema qui descritto ha il pregio di essere sufficientemente semplice da installare, ma richiede un esperto tecnico di manutenzione per rilevare eventuali tipi di malfunzionamento. E' possibile usare metodi elettronici più sofisticati (vedi il capitolo 11) per fornire un'indicazione su alcuni degli errori di funzionamento che si riscontrano più comunemente. Un vantaggio dell'uso della tecnologia elettronica avanzata descritta nel capitolo 11 è che il ritrovamento degli errori di funzionamento può essere semplificato.

5.4 Eliminatori dei rimbalzi di contatto

Si usano frequentemente interruttori come mezzo per fornire un segnale di ingresso ai sistemi logici. Nell'istante di chiusura dei contatti (o in quello della loro apertura) l'interruttore genera un "rumore" elettrico dovuto al "rimbalzo di contatto" che consiste in fluttuazioni ad alta frequenza nella tensione (e della corrente). Il rumore compare nel sistema logico come una serie di impulsi di spento-acceso-spento (Off-On-Off) a frequenza molto elevata; essi hanno un effetto limitato sui tipi di sistemi logici descritti fin qui nel libro, ma se un segnale "rumoroso" proveniente da un interruttore è applicato all'ingresso di un circuito contatore elettronico, il contatore memorizzerà il numero totale di impulsi di rumore invece di quello di effettiva apertura o chiusura dell'interruttore. Cosicchè, se si usa un interruttore per controllare, ad esempio il numero

di scatolette di fagioli su un nastro trasportatore, può darsi che il contatore elettronico riporti centinaia di scatolette tutte le volte che ne passa una in corrispondenza dell'interruttore. Dove un interruttore viene utilizzato per fornire i segnali di ingresso ad un circuito di conteggio, si usano appositi circuiti per eliminare le conseguenze del rimbalzo di contatto.

Due circuiti comuni di questo tipo sono riportati nella Fig. 5.5. Un sistema tipico utilizza le porte NOR ed è mostrato nella Fig. 5.5(a), nella quale le porte G1 e G2 formano un flip-flop *S-R*. Un interruttore a una via, due posizioni (deviatore) W è utilizzato come dispositivo di ingresso che applica un "1" logico alla linea d'ingresso appropriata. Quando i contatti dell'interruttore si chiudono, essi provocano ugualmente effetti di rimbalzo, ma il primo "1" logico applicato all'ingresso *S* provoca il posizionamento dell'uscita *Q* sull'"1" logico. Ulteriori impulsi di rimbalzo della tensione che si verificano prima della stabilizzazione finale non provocano nessun effetto aggiuntivo sullo stato dell'uscita *Q*.
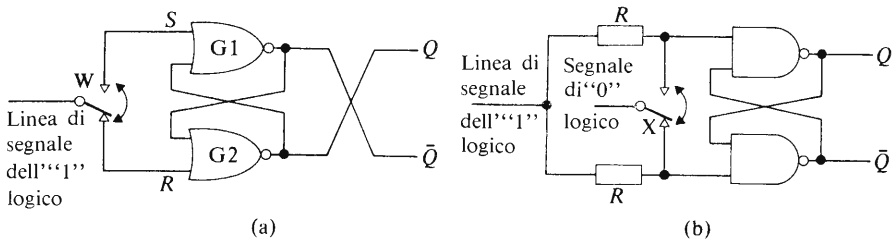
Fig. 5.5 Circuiti che eliminano il rimbalzo dei contatti

Nella Fig. 5.5(b), l'interruttore X applica un segnale corrispondente ad uno "0" logico per controllare il funzionamento del flip-flop NAND.

Uno svantaggio nell'uso degli interruttori elettrici come sensori, è rappresentato dall'ossidazione dei contatti: questa si verifica immancabilmente sui contatti degli interruttori e una soluzione spesso adottata per superare i suoi effetti consiste nell'usare un generatore di tensione che rompa lo strato di ossido; i valori tipici di queste tensioni variano tra 50 e 120 V. Poiché si tratta di tensioni molto più elevate di quelle impiegate nei circuiti logici, esse vengono ridotte ai livelli richiesti per mezzo di una semplice rete di attenuazione della tensione. Circuiti tipici che svolgono questa funzione sono riportati nella Fig. 5.6. L'inclusione di un condensatore C in questi circuiti elimina parzialmente il rimbalzo dei contatti, cosicché con si ha bisogno di circuiti come quello in Fig. 5.5

quando si utilizzano le soluzioni di Fig. 5.6.

Il circuito della Fig. 5.6(a) fa sì che venga applicato uno "0" al sistema logico quando l'interruttore è aperto e un "1" quando è chiuso. La resistenza R_1 e il condensatore C smorzano gli effetti di rimbalzo. In assenza di R_2, la tensione V_s è applicata ai terminali di ingresso del sistema logico; il valore di V_2 viene ridotto al di sotto del valore di V_s fino a raggiungere un valore adatto per il sisema logico, mediante l'inserimento della resistenza R_2. Il circuito della Fig. 5.6(b) fornisce invece un"1" quando l'interruttore è aperto e uno "0" quando è chiuso.
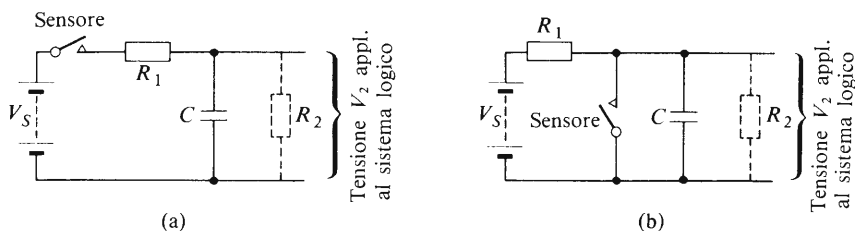
Fig. 5.6 Metodi per eliminare gli effetti dell'ossidazione dei contatti

5.5 Flip-flop master-slave

Alcuni progetti di circuiti contatori ad alta velocità possono portare a condizioni di "gara" simili a quelle descritte nel paragrafo 5.2 nel caso del flip-flop *S-R*. Per superare questo limite è stata sviluppata una famiglia di flip-flop noti come i flip-flop "master-slave". La base di molti di questi è il flip-flop master-slave *S-R*, descritto nel paragrafo 5.6. Altri tipi di flip-flop come i *J-K* e i *D* verranno descritti in paragrafi successivi.

5.6 Il flip-flop *S-R* master slave

Il principio del funzionamento del master-slave può essere compreso facendo riferimento alla Fig. 5.7.

Il circuito (che è invariabilmente un circuito integrato a causa della sua complessità) consiste di due flip-flop *S-R* in cascata, MFF (=master flip-flop) e SFF (= slave flip-flop), insieme coi quattro interruttori G1-G4 (che sono in realtà porte elettroniche). In aggiunta ai segnali di ingresso, di posizionamento e di riposizionamento (set e reset), viene applicato anche un terzo segnale di ingresso detto **segnale di clock** (orolo-
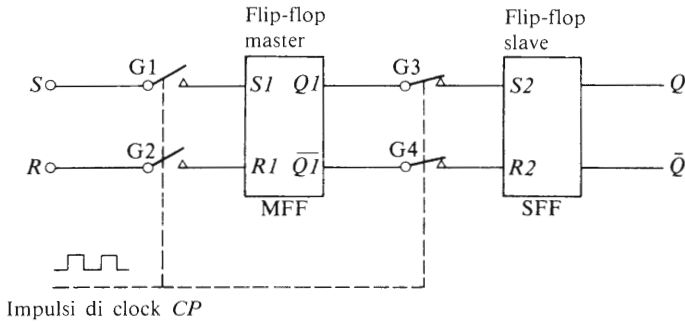
Fig. 5.7 La struttura fondamentale di un flip-flop *S-R*

gio). Quest'ultimo è generalmente un segnale ad onda quadra che oscilla tra i livelli dello "0" e dell'"1" (vedi il paragrafo 4.9). La funzione del segnale di clock è di controllare il flusso di informazione attraverso il flip-flop.

Una tipica forma d'onda, dell'impulso di clock è riportata nella Fig. 5.8 e mostra come le condizioni nel flip-flop siano correlate ai livelli logici dell'impulso di clock. Quando il segnale di clock è sullo "0" gli interruttori G1 e G2 sono aperti e quelli G3 e G4 sono chiusi. In questa condizione le linee d'ingresso (*S* e *R*) vengono isolate per mezzo di G1 e G2 dal flip-flop master MFF. Nello stesso tempo, le uscite di MFF sono collegate, tramite i contatti chiusi di G3 e G4, agli ingressi del flip-flop slave, SFF, cosicché l'uscita di SFF è uguale a quella di MFF. Cioè, $Q = Q1$ e $\overline{Q} = \overline{Q}1$. Queste condizioni si verificano nell'intervallo di tempo precedente l'istante t_A nella Fig. 5.8.

Allo stato logico "1" del segnale di clock tra gli istanti t_B e t_C in Fig. 5.8 corrisponde la chiusura degli interruttori G1 e G2 e l'apertura degli interruttori G3 e G4. Con ciò i segnali d'ingresso *S* e *R* vengono collegati al flip-flop MFF e contemporaneamente lo slave viene isolato dal master. In questo modo vengono immagazzinati *nuovi* dati nel master e i dati *vecchi* vengono conservati nello slave. Se l'uscita *Q* è rilevata con continuità su un indicatore di segnali logici, non verrà osservato alcun cambiamento sullo strumento, durante l'intero intervallo di tempo a sinistra dell'istante t_C in Fig. 5.8.

Quando il segnale di clock ritorna allo "0" logico (nell'istante t_D in Fig. 5.8), gli interruttori G1 e G2 si aprono e gli interruttori G3 e G4 si chiudono. Ancora una volta il master è isolato dagli ingressi *S* e *R* e l'uscita del master è collegata allo slave. In questo istante avviene che i

Fig. 5.8 Sequenza operativa di un flip-flop master slave in termini di impulso di clock

nuovi dati immagazzinati nel master vengono trasferiti nello slave: con questa azione si sostituiscono i dati vecchi nello slave con i nuovi dati.

Possiamo concludere perciò che **è necessario un intero impulso di clock per trasferire gli stati logici dai terminali di ingresso ai terminali di uscita.** Il ritardo di tempo per la realizzazione di questo trasferimento è uguale alla durata di un impulso di clock (che può essere di circa un microsecondo o ancora meno, a seconda della frequenza dell'oscillatore di clock). Possiamo perciò pensare che il trasferimento di dati avvenga in due stadi:

1. Quando l'impulso di clock cambia da "0" a "1" vengono trasferiti nuovi dati nel master (non si verifica alcun cambiamento nell'uscita Q).
2. Quando l'impulso di clock cambia da "1" a "0", i nuovi dati vengono trasferiti allo slave (e perciò all'uscita Q).

Si ricorda al lettore che se i dati "vecchi" e i dati "nuovi" sono identici, cioè, se $S=1$ e $Q=1$ o se $S=0$ e $Q=0$, non ci sarà nessuna variazione nell'uscita Q alla fine dell'impulso di clock. Soltanto nel caso in cui i dati vecchi e i dati nuovi siano diversi (per esempio $S=0$, $Q=1$), si verifica una variazione nell'uscita Q alla fine dell'impulso di clock.

La tabella della verità di un qualsiasi flip-flop è piuttosto complicata dato che dipende dallo stato precedente all'uscita. Per esempio, se lo stato iniziale dell'uscita Q fosse "0" e i nuovi ingressi fossero $S=1$ e $R=0$, l'uscita dopo un impulso di clock diventerebbe $Q=1$, cioè, Q

Tabella 5.2 Tabella della verità del flip-flop master slave *S-R*

Ingressi		*Uscite*	*Commento*
S_n	R_n	Q_{n+1}	
0	0	Q_n	Stato di memoria
0	1	0	Stato di riposizionamento
1	0	1	Stato di posizionamento
1	1	?	Stato incognito (non sappiamo)

cambierebbe stato. Tuttavia, se il valore iniziale di Q fosse 1 e i nuovi ingressi fossero $S=1$ e $R=0$, dopo un impulso di clock, l'uscita rimarrebbe invariata a $Q=1$. Supponiamo perciò che siano già stati applicati al flip-flop n insiemi di condizioni di ingresso, dove n è un qualsiasi numero intero, e che quindi si siano già verificati n insiemi di valori di Q.

Indichiamo l'ennesimo valore di Q con Q_n (che può essere o un "1" o uno "0"). A questo punto applichiamo i segnali di ingresso alle linee S e R che indicheremo con S_n e R_n ciascuna delle quali può essere un "1" o uno "0" — vedi Tabella 5.2).

Dopo il completamento del successivo impulso di clock (l'(n+1)-esimo), l'uscita assumerà la sua nuova condizione che indicheremo con Q_{n+1}. Il risultato è riportato nella colonna delle uscite della Tabella 5.2.

Vediamo che se $S=0$, $R=0$, $Q_{n+1}=Q_n$, cioè, lo stato di Q è inalterato dopo un impulso completo di clock; questo è lo stato di **memoria** del flip-flop.

Se $S=0$ e $R=1$, l'uscita di Q è riposizionata sullo "0" (reset) dopo un impulso completo di clock, quale che fosse la sua condizione originaria (che poteva essere un "1" o uno "0"); se $S=1$ e $R=0$, l'uscita è posizionata sull'"1" dopo un impulso completo di clock. Se $S=1$ e $R=1$, si stabilisce una condizione di gara nel flip-flop master quando il segnale di clock cade da "1" a "0", cosicché noi "non sappiamo" quale potrà essere la condizione di uscita, Q_{n+1}.

5.7 Il flip-flop master-slave *J-K*

Questo tipo di flip-flop è basato sul tipo master-slave *S-R*, con la differenza che ha dei collegamenti di retroazione tra l'uscita e l'ingresso (vedi Fig. 5.9a). L'effetto di questi collegamenti di retroazione è quello
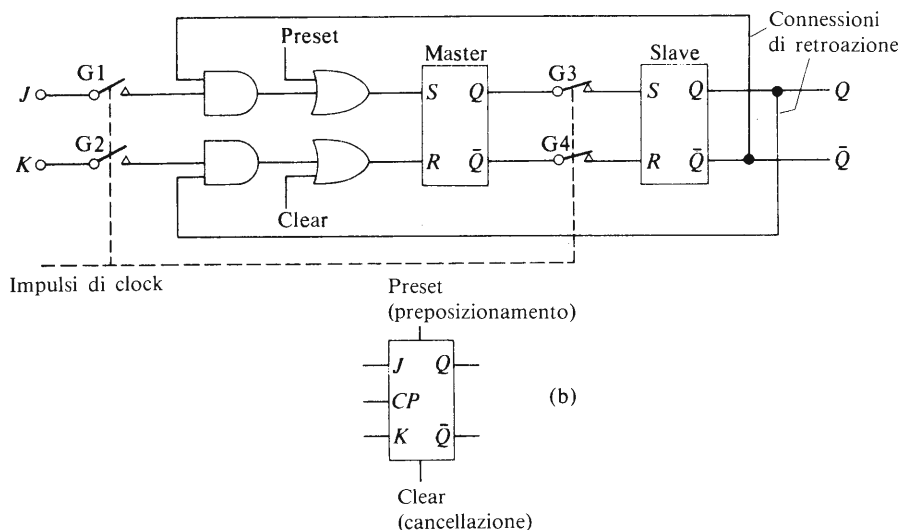
Fig. 5.9(a) Diagramma semplificato di un flip-flop master slave *J-K*; (b) simbolo circuitale per l'elemento *J-K*

di evitare le condizioni di "gara" che si verificano nel flip-flop *S-R* quando $S=1$ e $R=1$.

Gli interruttori G1, G2, G3 e G4 funzionano esattamente nello stesso modo descritto per il flip-flop master slave *S-R*; tuttavia in aggiunta alle linee d'ingresso (denominate *J* e *K* e *CP*, (per l'impulso di clock)), un effettivo elemento *J-K* ha due linee di controllo, denominate **preset e clear** (preposizionamento e cancellazione). La funzione di queste linee di controllo è quella di permettere all'utilizzatore di preposizionare (preset) l'uscita di *Q* sull'"1" logico, o di cancellare (clear) l'uscita *Q*, portandola allo "0" logico.

Si ottiene questo risultato applicando un "1" logico alla linea di controllo corrispondente (preset o clear) durante l'intervallo di tempo in cui il segnale di clock è nello stato di "0" logico. Il segnale logico "1" è applicato all'ingresso del flip-flop master tramite una porta OR; dato che gli interruttori G3 e G4 sono chiusi quando il segnale di clock è zero, lo stato del master è trasmesso direttamente allo slave. I segnali di preposizionamento e di cancellazione, perciò, preposizionano o cancellano simultaneamente entrambe le uscite *Q* del master e dello slave.

Il funzionamento del flip-flop *J-K* è descritto dalla sua tabella della verità dinamica in Tabella 5.3. I simboli J_n e K_n indicano i valori degli

ingressi J e K prima dell'(n + 1)-esimo impulso di clock e Q_{n+1} indica lo stato dell'uscita Q dopo l'(n + 1)-esimo impulso di clock.

Si può osservare che nelle prime tre righe della tabella della verità possiamo considerare un "1" logico applicato all'ingresso J (mentre K è 0) come equivalente al segnale: "posiziona Q nello stato "1"; e un "1" logico applicato all'ingresso K (mentre J è 0) come equivalente al segnale: "riposiziona Q sullo "0"". In questo senso, il flip-flop *J-K* svolge tutte le funzioni del flip-flop *S-R* con la convenzione che la linea J sostituisca la linea S e la linea K sostituisca la linea R.

Il funzionamento del flip-flop *J-K* si differenzia da quello del flip-flop *S-R* nell'ultima riga della Tabella 5.3.

Tabella 5.3 Tabella della verità del flip-flop *J-K*

J_n	K_n	Q_{n+1}	*Commento*
0	0	Q_n	Memoria
0	1	0	Riposizionamento
1	0	1	Posizionamento
1	1	Q_n	Commutazione

In questo caso, quando $J=1$ e $K=1$ ciascun impulso di clock provoca il cambiamento dello stato logico dell'uscita Q (cioè, $Q_{n+1} = \overline{Q}_n$) come segue. Supponiamo che Q_n sia 0; quando J è uguale a 1 e K è a sua volta uguale a 1, l'impulso successivo di clock fa sì che Q diventi "1"; l'impulso di clock ancora seguente fa sì che Q ritorni di nuovo a "0", e così via. Quindi partendo dalla condizione $Q_n = 0$, la sequenza degli stati di uscita sarà 0,1,0,1,0,1 ...; se, viceversa la condizione era $Q_n = 1$, la sequenza di valori di Q sarà 1,0,1,0,1 ... Quando entrambi gli ingressi J e K vengono posti nel livello logico "1", si dice che il flip-flop funziona nel **modo T** (*toggle*). Questo modo di funzionamento sta alla base di molti tipi di sistemi elettronici di conteggio (vedi capitolo 7).

5.8 Il flip-flop master-slave *D*

Il flip-flop D è così chiamato perchè può essere usato come elemento di ritardo (Delay) o come elemento di conservazione dei dati (Data latch).

Il flip-flop D ha una sola linea di ingresso dei segnali (la linea D) e

due uscite (Q e $\overline{Q}$). Può essere considerato come un elemento master-slave *J-K* avente la linea di ingresso *K* alimentata dal complementare logico del segnale di ingresso *J*, (vedi Fig. 5.10a). Perciò gli unici stati possibili sono: J (o D)$=1$ e $K=0$ oppure J (o F)$=0$ e $K=1$; lo stato di ingresso $J=0$ e $K=0$ come pure quello $J=1$ e $K=1$ non possono verifi-
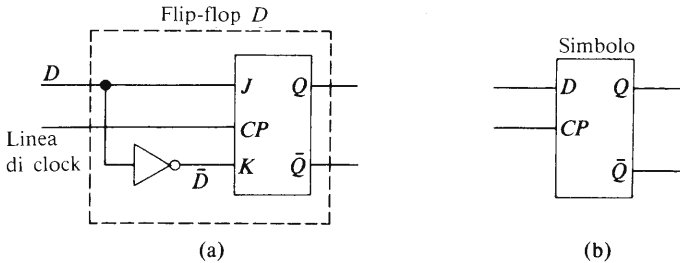
Fig. 5.10 Il flip-flop *D*: (a) circuito equivalente, (b) simbolo

carsi, dato che $J=D$ e $K=\overline{D}$. Il risultato globale è che la tabella della verità del flip-flop *D* è equivalente alla seconda e terza riga della tabella della verità del flip-flop *J-K* (vedi Tabella 5.3); poiché il flip-flop *D* ha solo una linea di ingresso (la linea *D*), la tabella della verità è quella riportata nella Tabella 5.4.

Tabella 5.4 Tabella della verità per il flip-flop *D*

D_n	Q_{n+1}
0	0
1	1

Molto semplicemente, la tabella della verità dice che se all'ingresso *D* è applicato uno "0", questo "0" subisce un ritardo di un intero impulso di clock prima di apparire all'uscita. Lo stesso ritardo temporale è subito da un segnale di ingresso corrispondente ad un "1" logico.

Dato che il flip-flop *D* ha una linea d'ingresso in meno rispetto al flip-flop *J-K*, il suo uso riduce la complessità dei sistemi. Il flip-flop *D* viene applicato per sostituire il flip-flop *J-K* in molti contatori.

5.9 Memorie statiche e dinamiche

Un elemento di **memoria statica** è un elemento nel quale possono essere immagazzinati dati per tutto il tempo in cui viene fornita tensione all'elemento stesso. Tutti i flip-flop descritti finora nel capitolo sono di questo tipo. Entrambe le tecnologie bipolari e MOS sono utilizzate nelle memorie statiche.

Un elemento di memoria dinamica è un elemento in cui i dati sono immagazzinati sotto forma di una carica elettrica su un condensatore. Se lo ''0'' logico corrisponde allo stato di condensatore scarico, allora l'''1'' logico corrisponde alla condizione in cui il condensatore è carico.

Il condensatore in questione è rappresentato generalmente dalla capacità tra la porta e il canale di un MOSFET. La carica immagazzinata da un condensatore svanisce gradualmente, cosicché una **memoria dinamica** conserva la sua quantità di informazione per un breve periodo di tempo: per evitare una perdita completa dei dati, l'informazione nella memoria è ''rinfrescata'' ogni 100 microsecondi circa per mezzo di circuiti elettronici aggiuntivi. Avremo occasione più avanti di occuparci delle memorie dinamiche.

Capitolo 6

Numerazione e aritmetica binaria

6.1. Sistemi di numerazione

Un sistema di numerazione è semplicemente un modo utile per indicare, per esempio, il valore del numero di componenti prodotti in un certo tempo. L'uomo ha trovato conveniente usare un sistema che opera in **base** dieci. Questo sistema è noto come sistema **decimale** e usa dieci cifre: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Il lettore rileverà il fatto che, in realtà, il numero zero è il **primo** numero della sequenza, e che il numero nove è il **decimo** numero. Ciascun sistema di numerazione deve contenere un numero che rappresenti lo zero, altrimenti sarebbe impossibile dire che la quantità di oggetti in un certo sistema è zero.

Dato che i sistemi logici sono dispositivi "acceso-spento" (ON-OFF) e che le loro uscite possono assumere solo il valore "1" logico e "0" logico, essi sono adattissimi ad operare con un sistema di numerazione in base 2. Questo sistema è noto come **sistema binario**. E' pratica comune che, quando si usano sistemi di numerazione in basi diverse, si indichi la base scrivendola come suffisso in basso alla destra del numero, come per esempio:

$$897 \text{ decimale} = 897_{10}$$
$$1011 \text{ binario} = 1011_{2}$$

Un sistema di numerazione assai usato nei microprocessori (vedi anche capitolo 11), è il sistema di numerazione **esadecimale** (abbreviazione **hex**), la cui base contiene sedici elementi (i numeri tra 0 e 15). La Tabella 6.1 riporta un elenco dei primi 24 numeri nei sistemi decimali, binario e esadecimale.

Per spiegare come vengono costruite tabelle di questo genere facciamo riferimento alla familiare sequenza decimale. Partendo dal numero zero, vediamo che quando raggiungiamo il 9_{10} abbiamo "consumato" tutte le cifre disponibili. Il numero appena successivo, perciò diventa "una volta

Tabella 6.1

Sistema	*Decimale*	*Binario*	*Esadecimale*	*Decimale*	*Binario*	*Esadecimale*
Base	10	2	16	10	2	16
	0	0	0	12	1100	c
	1	1	1	13	1101	d
	2	10	2	14	1110	e
	3	11	3	15	1111	f
	4	100	4	16	10000	10
	5	101	5	17	10001	11
	6	110	6	18	10010	12
	7	111	7	19	10011	13
	8	1000	8	20	10100	14
	9	1001	9	21	10101	15
	10	1010	a	22	10110	16
	11	1011	b	23	10111	17

dieci più zero unità'', o 10_{10}; il numero successivo al 10_{10} è quindi ''una volta dieci più una unità'', cioè 11_{10}, ecc.

Dato che la base del sistema binario è 2, ci troviamo ad aver consumato tutte le cifre disponibili quando abbiamo scritto 0_2 e 1_2. Il numero appena successivo è perciò ''una volta due più zero unità'' cioè 10_2 (che viene pronunciato ''uno, zero'' e non ''dieci''); questo numero è equivalente al numero due in base dieci. Il numero successivo è ''una volta due più una unità'' cioè: 11_2 $(=3_{10})$. Dato che abbiamo di nuovo consumato tutte le possibilità su entrambe le colonne (si ricordi che solo le cifre 0 e 1 possono venir usate nel sistema binario), il numero che segue è ''una volta quattro più zero volte due più zero unità'' cioè 100_2 $(=4_{10})$. Cosicchè il valore di ciascuna colonna nel sistema binario è doppio di quello della colonna precedente.

Possiamo perciò affermare che:

$$12_{10} = (1 \times 8) + (1 \times 4) + (0 \times 2) + (0 \times 1) = 1100_2$$

Nel sistema esadecimale, vengono usate le cifre decimali dallo zero al nove, dopo le quali dobbiamo inventarne alcune nuove per rappresentare i numeri equivalenti a quelli decimali dal dieci al quindici (si ricordi che ci sono sedici cifre nel sistema esadecimale). I ''numeri'' scelti sono rappresentati dalle prime sei lettere dell'alfabeto. Perciò $A_{16} = 10_{10}$,

$B_{16} = 11_{10}$ ... $F_{16} = 15_{10}$. Quando raggiungiamo F_{16} abbiamo consumato tutti i caratteri del sistema esadecimale, cosicché il numero appena successivo è "una volta sedici più zero unità", cioè 10 nel sistema esadecimale corrisponde al 16 nel sistema decimale. Il numero ancora successivo è "una volta sedici più una unità" cioè 11_{16} (equivalente a 17 nel sistema decimale), ecc.

6.2 Immagazzinamento di numeri frazionari

Ciascuna cifra in un numero decimale alla sinistra della "virgola" decimale è associata ad una "potenza" di dieci con esponente positivo, e ciascuna cifra alla destra della virgola decimale è associata ad una potenza di dieci con esponente negativo, come segue:

$$98{,}76_{10} = (9 \times 10^1) + (8 \times 10^0) + (7 \times 10^{-1}) + (6 \times 10^{-2})$$
$$= (9 \times 10) + (8 \times 1) + (7 \times 0{,}1) + (6 \times 0{,}01)$$

Il punto decimale separa perciò la parte del numero maggiore dell'unità da quella minore dell'unità.

Nel sistema binario ciascuna cifra binaria (**B**inary dig**IT**, abbreviata *bit*) è associata col numero due elevato ad un esponente appropriato, come è mostrato dall'esempio che segue:

$$10{,}01_2 = (1 \times 2^1) + (0 \times 2^0) + (0 \times 2^{-1}) + (1 \times 2^{-2})$$
$$= (1 \times 2) + (0 \times 1) + (0 \times 1/2) + (1 \times 1/4)$$

In questo caso la *virgola binaria* separa la parte del numero maggiore dell'unità dalla parte che è minore dell'unità.

Per questo, un sistema digitale con una capacità di immagazzinamento di quattro bit può essere utilizzato per immagazzinare numeri che variano tra 0000_2 e 1111_2 ($= 15_{10}$) a intervalli di una unità, oppure in alternativa a intervalli di $0{,}01_2$ ($= 1/4$) da 00,00 a $11{,}11_2$($= 3\frac{3}{4}$) o ancora a intervalli di $0{,}0001_2$ ($= 1/16$) da 0,0000 a 0,1111 ($= 15/16$), ecc.

6.3 La lunghezza delle "parole" (words) binarie

I dati sotto forma di valori numerici o di caratteri alfabetici sono immagazzinati nei sistemi digitali sotto forma di **parole** (*words*) binarie.

Una parola binaria consiste in una sequenza di zero e di uno che può essere decodificata in una forma comprensibile dall'uomo per mezzo di

caratteri alfabetici propri o di numeri decimali.

Il numero totale di bit di ciascuna parola viene indicato come **lunghezza della parola** (*word length*); le lunghezze tipiche delle parole si trovano in un intervallo compreso fra 4 e 64 bit.

Ciascuna posizione nella parola deve essere riempita da una cifra binaria, in modo tale che, nel caso si avesse il numero 0 in una parola particolare, tutte le posizioni della parola dovrebbero essere riempite da zeri.

Per esempio, si consideri un sistema che abbia una lunghezza di otto bit, nel quale vogliamo immagazzinare numeri binari corrispondenti ai valori decimali 8,0, $6\frac{1}{8}$, $12\frac{1}{16}$. Per convenienza mettiamo la *virgola binaria* al centro della parola.

Numero	*Parola binaria*
8	1000,0000
0	0000,0000
$6\frac{1}{8}$	0110,0010
$12\frac{1}{16}$	1100,0001

Quando scriviamo un numero dobbiamo sempre partire dal presupposto che entrambe le posizioni a destra e a sinistra del numero siano riempite di zeri. Per esempio, quando scriviamo il numero decimale 8 non scriviamo normalmente 0008,0000.

Tuttavia, quando usiamo un sistema digitale, è necessario che tutte le posizioni contengano un valore binario (che può essere 0 o 1) in modo tale da "riempire" completamente la parola.

Per esempio, quando immagazziniamo il numero $6\frac{1}{8}$ dobbiamo immagazzinare il valore 0110,0010 se la lunghezza di ogni parola è di otto bit. Gli zeri addizionali che vengono immagazzinati alle estremità del numero semplicemente per riempire le posizioni inutilizzate nel numero, vengono designati col nome di **zeri non significativi**.

6.4 Codici decimali codificati in binario (BCD)

Dato che il codice decimale è compreso dall'uomo e il codice binario è "capito" dai sistemi logici, è stata sviluppata una serie di codici noti come **decimali codificati in binario** (*binary-coded decimal* o BCD) che permettono un libero scambio di informazioni tra i due codici.

Per trasmettere valori decimali abbiamo bisogno di dieci insiemi di se-

quenze diverse e identificabili che corrispondono alla serie dallo 0_{10} al 9_{10}. Possiamo **codificare** i dieci numeri in forma binaria utilizzando una sequenza binaria di quattro bit (quattro cifre binarie). I quattro bit danno $2^4 = 16$ combinazioni possibili; dieci di queste vengono usate per i numeri decimali, mentre sei non sono richieste e vengono dette **ridondanti**.

Ci sono molte forme possibili di codici BCD; tre di questi sono riportati nella Tabella 6.2. A ciascuna colonna del codice viene attribuito un peso decimale e se sommiamo i "pesi" corrispondenti ai vari uno che vi compaiono, otteniamo il corrispondente valore nel sistema decimale. Per

Tabella 6.2 Alcuni codici BCD

Valore decimale	*Codici BCD*		
	8421	2421	7421
0	0000	0000	0000
1	0001	0001	0001
2	0010	0010	0010
3	0011	0011	0011
4	0100	0100	0100
5	0101	0101	0101
6	0110	0110	0110
7	0111	0111	1000
8	1000	1110	1001
9	1001	1111	1010

esempio, nel codice BCD 8421, il "peso" di un "1" nella colonna di sinistra è quello di un 8 decimale mentre un "1" nella colonna di destra ha un peso equivalente a quello di un 1 decimale. Quindi il numero decimale 9 è rappresentato nel codice BCD 8421 da 1001, cioè $(1 \times 8) + (0 \times 4) + (0 \times 2) + (1 \times 1)$; nel codice BCD 2421 il numero 9_{10} è rappresentato dal numero 1111, cioè $(1 \times 2) + (1 \times 4) + (1 \times 2) + (1 \times 1)$; nel codice BCD 7421 il numero 9_{10} appare come 1010, cioè $(1 \times 7) + (0 \times 4) + (1 \times 2) + (0 \times 1)$. Sebbene ci siano molti codici BCD, quando si fa riferimento "al" codice BCD senza accennare esplicitamente ai pesi, viene inteso che si parla del codice BCD 8421.

La scelta di un codice particolare dipende dalla sua utilità in una particolare applicazione; è perciò impossibile dire se esiste un codice "ottimo" per tutte le applicazioni.

I numeri superiori al 9_{10} vengono rappresentati nel codice BCD ripe-

tendo i gruppi in BCD per ciascuna cifra; per esempio i pesi nel gruppo delle unità del codice BCD 8421 sono 8421; i pesi nel gruppo delle decine sono 80, 40, 20, 10; nel gruppo delle centinaia sono 800, 400, 200, 100, ecc. Per esempio il numero decimale 187,6 è trascritto nel codice BCD 8421 nella forma 0001 1000 0111, 0110.

6.5 Codici alfanumerici

Molti sistemi digitali trattano informazioni sia alfabetiche che numeriche. Per trasmettere un testo completo in lingua inglese, anche nell'ipotesi di trasmettere solo lettere maiuscole, abbiamo bisogno di almeno 36 "parole" codificate diversamente; queste vanno usate per codificare le 26 lettere dalla A alla Z e le dieci cifre dallo 0 al 9. In aggiunta dobbiamo poter trasmettere un certo numero di segni di interpunzione (punto, virgola, spazio tra le parole, ecc.), ciascuno dei quali richiede una parola in codice, insieme alle istruzioni per la telescrivente (per esempio il comando che fa tornare indietro il carrello all'inizio della linea quando la linea precedente è completa, oppure il comando che fornisce la carta ad una nuova linea quando deve essere cominciata). Per questa ragione abbiamo bisogno di almeno un codice di 6 bit che fornisca $2^6 = 64$ parole codificabili per trattare anche il più elementare testo in lingua inglese (un codice con parole di 5 bit fornisce soltanto $2^5 = 32$ parole codificabili, che non sono sufficienti per codificare nemmeno tutte le lettere maiuscole dell'alfabeto e le cifre decimali).

Tabella 6.3 Un codice alfanumerico

Numero decimale	*Codice*	*Lettere*	*Codice*	*Lettere*	*Codice*	*Lettere*	*Codice*
0	00 000	*A*	01 0000	*K*	10 0000	*U*	11 0000
1	00 0001	*B*	01 0001	*L*	10 0001	*V*	11 0001
2	00 0010	*C*	01 0010	*M*	10 0010	*W*	11 0010
3	00 0011	*D*	01 0011	*N*	10 0011	*X*	11 0011
4	00 0100	*E*	01 0100	*O*	10 0100	*Y*	11 0100
5	00 0101	*F*	01 0101	*P*	10 0101	*Z*	11 0101
6	00 0110	*G*	01 0110	*Q*	10 0110		
7	00 0111	*H*	01 0111	*R*	10 0111		
8	00 1000	*I*	01 1000	*S*	10 1000		
9	00 1001	*J*	01 1001	*T*	10 1001		

In Tabella 6.3 è riportata una forma di codice alfanumerico a 6 bit. I quattro bit a destra di questo codice seguono il codice BCD 8421, i due bit di sinistra sono usati per identificare i numeri decimali e i gruppi di lettere dell'alfabeto.

Uno dei codici di trasmissione delle informazioni più comunemente usato è il codice ASCII (American Standard Code for Information Interchange): esso usa una parola di 8 bit: sette bit forniscono un totale di 128 caratteri ai fini della trasmissione delle informazioni, mentre l'ottavo bit fornisce un controllo sull'accuratezza delle informazioni codificate dagli altri sette bit (vedi paragrafo 6.6).

6.6 Rilevazione degli errori mediante l'uso del controllo della parità (parity checks)

Quando si trasmettono informazioni in forma digitale può accadere che il segnale ricevuto sia diverso dal segnale trasmesso. Questo fatto può derivare da un gran numero di cause, tra le quali le tensioni indotte nelle linee di trasmissione dovute a interferenze elettriche locali. Se il segnale è trasmesso via radio possono venire create interferenze da altri trasmettitori o da effetti atmosferici.

Questi segnali di "rumore" o di interferenza possono far sì che un gruppo di cifre codificate contenga un "1" dove dovrebbe esserci uno "0" e viceversa. In alcuni casi l'informazione può essere registrata su nastro magnetico (caso che si verifica per esempio per alcune macchine utensili con controllo numerico); difetti del nastro magnetico o nel sistema di registrazione, danno luogo al tipo di errori appena descritto.

La forma più semplice di rilevamento degli errori consiste nell'aggiungere un bit noto come il **bit di parità** (parity bit) ad ogni parola binaria. (Esso è ridondante per quello che concerne la trasmissione delle informazioni). Ci sono due tipi di controllo della parità noti rispettivamente come "parità dispari" e "parità pari". Se usiamo il metodo di controllo delle **parità dispari** (odd parity) il bit di parità (il bit P nella Tabella 6.4) è "0" se il numero decimale che esprime la somma degli "1" contenuti nella parola codificata è dispari, ed è "1" se la somma degli "1" è pari.

Se utilizziamo il metodo di controllo della **parità pari** (even parity) il bit di parità è "0" se il numero decimale che esprime la somma degli "1" nella parte della parola adibita alla trasmissione delle informazioni è pari, ed è un "1" se la somma degli "1" è dispari.

Un bit di parità usato nella maniera appena descritta fornisce un modo rapido di controllare l'esistenza di un *singolo errore*, ma non può correggerlo. Per esempio se la parola binaria trasmessa con il bit di pari-

Tabella 6.4 Esempi di controlli di parità

Parità dispari Dati	P	*Parità pari* Dati	P
0000	1	0000	0
0010	0	0010	1
0011	1	0011	0
1011	0	1011	1
1111	1	1111	0

tà dispari è 00111 e la parola ricevuta è 10111, l'apparecchio ricevente può rilevare l'errore ma non può correggerlo. Inoltre, se vengono commessi due errori nella stessa parola, il controllo di parità semplice non può rilevare l'errore.

Per poter rilevare l'errore in una parola del codice e *correggerlo*, è necessario usare più di un bit ridondante.

Sono stati sviluppati molti codici sofisticati per correggere gli errori; per esempio, un tipo di codice che usa parole di 4 bit per trasmettere i dati ha una lunghezza totale della parola di 15 bit; gli undici restanti sono bit ridondanti per il controllo.

Un *codice che corregge gli errori* abbastanza semplicemente è il **codice del blocco di parità**, illustrato nella Tabella 6.5. Questo codice ha un bit di parità P_l nella posizione di estrema destra di ciascuna parola (contenente altri quattro bit per trasmettere informazioni), ciascuno dei quali fornisce un controllo della parità su una linea. Ogni quattro parole c'è una parola di parità P_w, ciascun bit della quale fornisce un controllo della parità su una colonna di quattro bit.

Tabella 6.5 Esempio di controllo per mezzo del blocco di parità

	Informazione				P_l	
Informazione	0	1	0	1	1	
	0	0	0	0	1	Questo "1" dovrebbe essere uno "0"
	1	0	1 ←	1	1 ←	Errore in questa riga
	1	0	0	0	0	
P_w	1	0	1	1	0	
			↑			Errore in questa colonna

In questo modo, quando vogliamo trasmettere quattro parole contenenti informazioni per quattro bit, è necessario trasmettere cinque parole di cinque bit nella forma del blocco riportato in Tabella 6.5: in essa è utilizzato il controllo della parità dispari. Controllando ciascuna linea si nota che nella terza c'è un errore di parità; controllando ciascuna colonna si nota che, nella terza c'è un errore. L'errore si trova perciò all'intersezione di questa linea e di questa colonna ed è il terzo bit della terza parola; viene corretto sostituendo uno "0" al posto dell'"1" esistente.

6.7 Codici sensibili alla posizione

La posizione di un oggetto, per esempio la slitta di una macchina utensile, può essere misurata mediante l'uso di un piatto o di un disco codificato nel sistema binario. Un esempio di piatto codificato col sistema binario è riportato in Fig. 6.1. Nella figura le aree ombreggiate rappresentano le regioni isolanti e le aree chiare rappresentano le regioni conduttrici. Se il piatto è assicurato alla slitta della macchina utensile, mentre questa si muove longitudinalmente le luci danno una indicazione binaria della sua posizione.
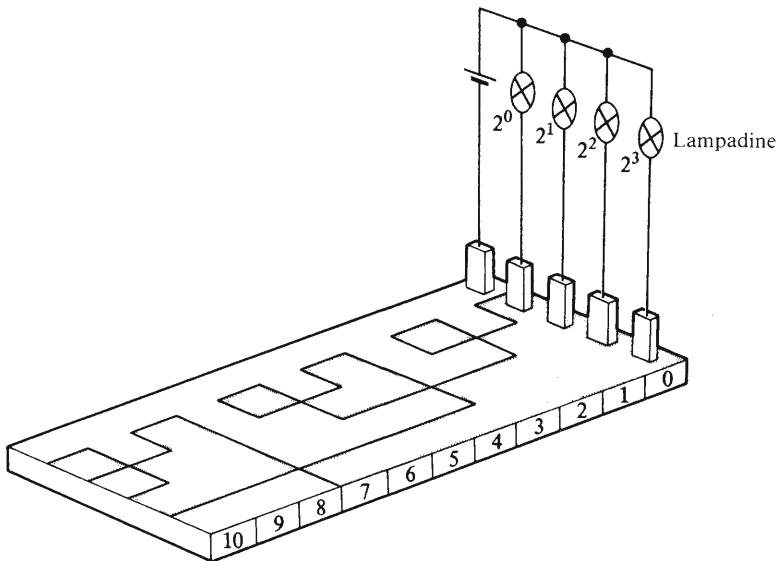
Fig. 6.1 Diagramma di un piatto codificato in binario per la rilevazione della posizione

Supponiamo che il carrello si trovi nella posizione "7" e si stia muovendo verso la posizione "8". Inizialmente le lampadine accese forniscono il numero binario 0111 (corrispondente a 7_{10}), e quando la transizione verso la nuova posizione è completata, essi forniscono il numero binario 1000 (corrispondente a 8_{10}).

Perché questo cambiamento avvenga, *tutti i quattro bit devono cambiare stato simultaneamente.*

A causa delle difficoltà meccaniche per allineare le spazzole ed anche a causa di problemi creati dall'usura fisica, è quasi impossibile che tutti i bit cambino simultaneamente in un prodotto di serie. Come conseguenza è praticamente certo che soltanto una lampadina alla volta cambierà il suo stato. Una possibile sequenza di eventi è riportata nella Tabella 6.6.

Tabella 6.6

	Codice binario	*Equivalente decimale*
Posizione iniziale	0111	7
Uscite possibili durante la transizione	0110	6
	0010	2
	1010	10
Posizione finale	1000	8

In dipendenza del tipo di sequenza di cambiamenti che si verifica, può essere formato un qualsiasi numero binario compreso tra 0000 e 1111 durante il periodo di transizione. Chiaramente codici di questo tipo sono inadatti al rilevamento della posizione a causa degli errori nell'indicazione che possono verificarsi durante i periodi di transizione.

Ciò che serve per un'applicazione di questo tipo è un codice nel quale cambi un solo bit quando ci si sposta da una posizione qualsiasi nella sequenza del codice a quella immediatamente adiacente (in entrambe le direzioni). I codici di questo tipo sono noti come **codici a distanza unitaria** (unit distance codes). Uno dei più comunemente usati è il **codice Gray**; un codice Gray a 4 bit è riportato in Tabella 6.7. Un'analisi della Tabella 6.7 mostra che il codice Gray è un **codice non pesato**, dato che è impossibile dare un peso decimale a ciascuna colonna.

I lettori noteranno che l'ultimo gruppo codificato (15_{10}) del codice Gray è 1000 e che questa codifica richiede che si cambi in "0" solo il primo bit a sinistra per ritornare alla partenza della sequenza (0000). Il

Tabella 6.7 Il codice Gray a 4 bit

Valore decimale	*Codice Gray*	*Valore decimale*	*Codice Gray*
0	0000	8	1100
1	0001	9	1101
2	0011	10	1111
3	0010	11	1110
4	0110	12	1010
5	0111	13	1011
6	0101	14	1001
7	0100	15	1000

cambiamento unitario di distanza

codice in Tabella 6.7 può essere usato quando il movimento complessivo può essere suddiviso in 16 incrementi. Altri codici a distanza unitaria possono venire elaborati per altre suddivisioni degli spostamenti, per esempio per dieci spostamenti (decimale). Tuttavia una trattazione più dettagliata di questi codici esula dagli scopi di questo libro.

6.8 Addizione binaria

Dato che possiamo avere solo due valori (0 e 1) nel sistema binario, quando eseguiamo un'addizione binaria (1 + 1) diciamo

$$1+1=\text{zero e ``riporto uno''}$$
$$=0 \text{ riporto } 1$$

Questa operazione ha il suo equivalente nel sistema decimale quando diciamo $1+9=$ zero e "riporto uno". Nell'analisi precedente il simbolo "+" è, naturalmente, il simbolo "più" di addizione aritmetica e non rappresenta la funzione logica OR.

La tabella della verità dei processi elementari di addizione è riportata in Tabella 6.8.

Un circuito logico per questa parte del processo può essere dedotto dalla tabella della verità come segue. Nella colonna della "somma" otteniamo un'uscita logica corrispondente all'"1" logico quando

$$S=(\text{NOT } A \text{ AND } B) \text{ OR } (A \text{ AND NOT } B)=\overline{A}.B+A.\overline{B}$$

Tabella 6.8 Addizione binaria

Numeri da addizionare		*Somma*	*Riporto*	*Commento* ("+" = simbolo di addizione aritmetica)
A	*B*	*S*	*C*	
0	0	0	0	0+0 = 0
0	1	1	0	0+1 = 1
1	0	1	0	1+0 = 1
1	1	0	1	1+1 = 0 riporto 1

Troviamo un "1" nella colonna del "riporto" della Tabella 6.8 quando

$$C = A \text{ AND } B = A.B$$

Usiamo le tecniche sviluppate nel capitolo 2 e costruiamo il diagramma logico a blocchi riportato nella Fig. 6.2(a). Le porte G1, G2, ···G5 compresa, formano un circuito di "SOMMA" e la porta G6 è il circuito
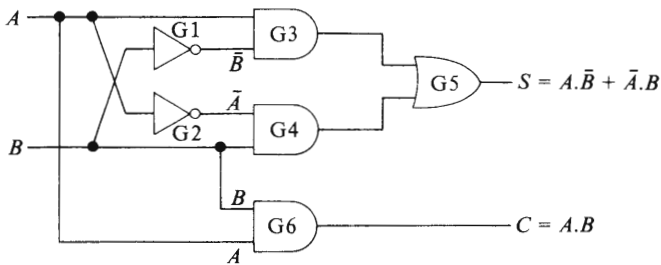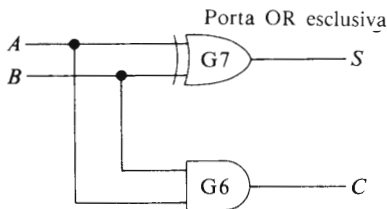
Fig. 6.2 Diagramma logico a blocchi di un semiaddizionatore (half-adder)

di "RIPORTO". La parte "SOMMA" del circuito può essere sostituita da una singola porta, conosciuta sotto il nome di porta OR-ESCLUSIVA o porta NOT-EQUIVALENTE simboleggiata dalla porta G7 nella Fig. 6.2(b). L'espressione logica sviluppata da questa porta è $\bar{A}.B+A.\bar{B}$.

Entrambi i circuiti della Fig. 6.2 sono noti come circuiti *semiaddizionatori* per la seguente ragione. Quando sommiamo due numeri otteniamo la somma e il riporto dai due numeri dati; il riporto deve essere "portato" avanti e addizionato al valore immediatamente seguente nel processo di addizione. In questo modo il processo di **addizione completa** coinvolge non solo l'addizione dei valori di A e B ma anche l'addizione del bit che viene "riportato" dal risultato dell'addizione precedente. Una rete che sia capace di addizionare tutte e tre queste quantità è nota come **addizionatore completo** (full-adder).

Per illustrare il processo di addizione completa, consideriamo l'addizione dei numeri binari 11,0 e 11,0 ($3_{10}+3_{10}$) come segue:

Ingressi	Numero A	0	1	1	,	0
	Numero B	0	1	1	,	0
	Riporto C_1	1	1	0		
Uscite	Somma S	0	1	1	,	0
	Riporto C_0	1	1	0	,	0

La colonna più a destra, corrispondente al bit *meno significativo*, viene sommata per prima per dare $0+0=0$ riporto 0; il *riporto* C_0 ottenuto da questa addizione è perciò zero. Esso è trasferito nella linea del riporto C_1 della terza colonna. In questa colonna abbiamo l'addizione $1+1+1=1$ riporto 1. Il riporto C_1 di questa colonna viene quindi trasferito nella linea del riporto C_1 della colonna finale (*la più significativa*) dove abbiamo la somma $0+0+1=1$ riporto 0.

Un diagramma a blocchi di un tipo di circuito addizionatore completo è riportato in Fig. 6.3. Il circuito ha tre ingressi, A e B (i valori da addizionare) e il riporto C_1 proveniente dall'addizione precedente. Tale circuito ha due uscite: la **somma** S_0, che fornisce la somma binaria di A, B, e C_1, e il **riporto** C_0, che fornisce il bit di "riporto" per l'addizione della "colonna" successiva.
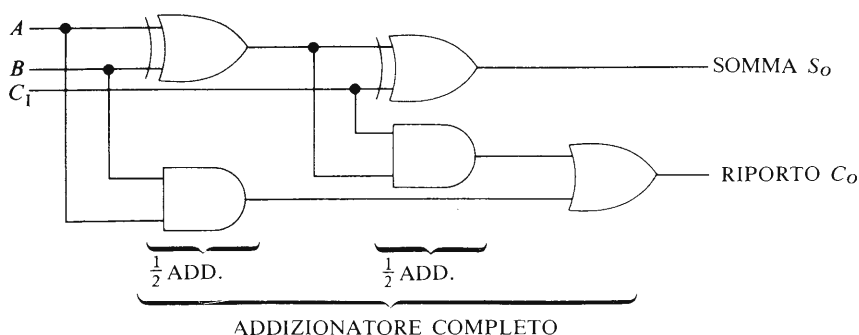
Fig. 6.3 Diagramma a blocchi di un tipo di addizionatore completo

6.9 Sottrazione binaria

Quando si sottrae il numero decimale cinque dal numero decimale otto scriviamo semplicemente $8-5=3$. Sfortunatamente il concetto di "segno negativo" è interamente una creazione umana, ed è difficile insegnare ad un sistema digitale la differenza tra "più" e "meno". Per rendere un sistema digitale capace di sottrarre i numeri, è stato creato un metodo per informare il sistema digitale che sta trattando con numeri negativi.

Un metodo per immagazzinare questa informazione è quello di includere una cifra binaria addizionale nella posizione più significativa della parola digitale, cioè nel bit più a sinistra della parola. Assumiamo per il momento di aver a che fare con una lunghezza di "parola" di cinque bit, il cui bit più significativo è il cosiddetto **bit di segno** (sign bit). Il valore di questo bit è generalmente "0" per i numeri positivi ed è "1" per i numeri negativi. Per facilitare i lettori, nel seguito metteremo il bit di segno tra parentesi; quando è immagazzinato in un sistema digitale, il bit di segno è identificato semplicemente dal fatto che occupa la posizione più significativa del numero.

Riportiamo qui di seguito alcuni esempi tipici:

$$+8_{10} = (0)1000_2$$
$$+5_{10} = (0)0101_2$$
$$-3_{10} = (1)0011_2$$
$$-9_{10} = (1)1001_2$$

Lo svantaggio principale del metodo appena descritto di rappresentazione dei numeri (noto come il metodo del *segno e modulo*) è che dob-

biamo tuttora eseguire la sottrazione in binario, il che significa che abbiamo bisogno di un circuito di sottrazione che differisce dal circuito di addizione.

Le cose migliorano, tuttavia, quando possiamo "utilizzare" il circuito di addizione per eseguire la nostra sottrazione. Questo è realizzabile utilizzando il procedimento cosiddetto della **addizione del complemento.** Useremo nel seguito la notazione del cosiddetto **complemento a due.** L'equivalente "negativo" di un qualsiasi numero è ottenuto scrivendo il suo complemento a due secondo la regola seguente.

Si copia il numero binario cominciando dalla cifra meno significativa fino all'uno meno significativo compreso; si ha quindi il complemento di tutte le altre cifre (gli zeri vengono cambiati in uno e gli uno in zero).

Per esempio, facciamo il complemento a due del numero binario (0)10,010 ($+2{,}25_{10}$).

$$\text{complemento a 2 di } (0)10{,}010 = \underbrace{(1)01{,}1}_{\text{invertiti}}\underbrace{10}_{\text{invariati}}$$

Ulteriori esempi della regola appena data vengono illustrati di seguito.

Numero binario	Complemento a due
(0)111,0	(1)001,0
(0)100	(1)100
(0)000,01	(1)111,11
(1)001,10	(0)110,10
(1)111	(0)001

Il lettore noterà che gli ultimi due valori binari sono essi stessi numeri negativi e dato che il "negativo" di un numero negativo è un numero positivo, il risultato dell'operazione di complemento a due è numero positivo (indicato dal bit di segno nella colonna del complemento a due). Il lettore dovrebbe rilevare che quando si immagazzinano numeri binari sotto forma di complemento a due, se il bit di segno è zero, allora la parte della parola riservata alle informazioni è il valore reale e ha segno positivo. Se il bit di segno è uno, allora la parte della parola riservata alle informazioni è il complemento a due del numero.

Mostreremo il procedimento di sottrazione mediante *l'addizione del*

complemento a due come segue. Supponiamo di voler sottrarre il numero decimale 4 dal numero decimale 9 usando il sistema di numerazione binaria.

Ora $+4_{10} = (0)0100_2$
Quindi $-4_{10} = (1)1100$ complemento a due

Per sottrarre quattro da nove, *addizioniamo* il complemento a due di quattro all'equivalente binario di nove come segue:

$+9_{10}$	= (0)1001	(valore binario puro)
-4_{10}	= (1)1100	(complemento a due)
somma perso ←	1(0)0101	risultato dell'addizione

Dato che la lunghezza totale della parola o la capacità di immagazzinare del nostro sistema digitale è di cinque bit (compreso il bit di segno), il bit "1" che "supera la capacità" è "perso" per il sistema. Questo è equivalente alla condizione del contachilometri della macchina quando superiamo la capacità totale del contachilometri stesso; se il contachilometri può indicare un numero di chilometri fino a 99999, quando percorriamo il chilometro successivo, esso indica 00000 e l'uno "straripato" (straripare = to over flow) è "perso". Il risultato della sottrazione precedente è perciò

$$(0)0101_2 = +5_{10}$$

Il lettore potrà verificare le conoscenze accumulate finora sottraendo 9 da 4 per mostrare che il risultato binario è (1)1011 (cioè -5_{10}).

6.10 Moltiplicazione

La moltiplicazione è una forma di addizione ripetuta, cosicché il calcolo di 53 × 28 può essere eseguito addizionando 53 a se stesso 27 volte. Questo metodo è troppo lento per essere utilizzato nei calcolatori, ma serve a mostrare come un circuito di addizione può essere usato anche per la moltiplicazione.

6.11 Divisione

La divisione è una forma di sottrazione ripetuta. Per esempio per dividere 14 per 5, sottraiamo 5 da 14 fino a quando il numero che rimane è minore di cinque. In questo caso il numero di sottrazione dà un quoziente di due con resto di quattro. Questo processo è a sua volta troppo lento per essere utilizzato in un sistema pratico, ma mostra che un circuito di sottrazione (o di addizione del complemento) può essere utilizzato per questo scopo.

Capitolo 7

Circuiti contatori

I contatori sono utilizzati in applicazioni che vanno dal computo degli stipendi alla registrazione del numero degli scatti telefonici. Il tipo di circuito contatore utilizzato dipende da un insieme di fattori che comprendono il tipo di codice binario impiegato e la velocità di funzionamento del sistema.

7.1 A che velocità?

Molti sistemi utilizzati oggi funzionano a velocità moderate e possono perciò impiegare una circuiteria relativamente semplice. Un esempio di applicazione di contatore è quello della registrazione del numero di componenti che passano lungo un nastro trasportatore (vedi Fig. 7.1).

In questo tipo di contatore, la velocità con cui si muovono i componenti lungo il nastro trasportatore è relativamente bassa, cosicché è sufficiente un sistema di conteggio a bassa velocità.
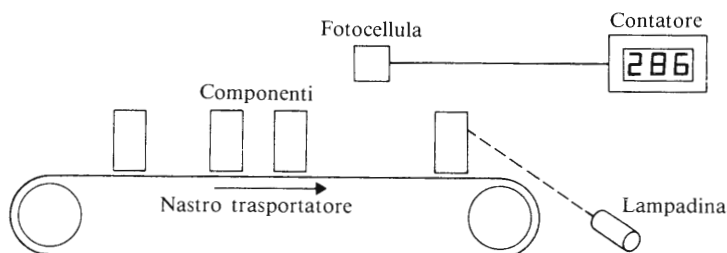
Fig. 7.1 Conteggio dei componenti lungo la linea di produzione

Se dobbiamo misurare la velocità di un proiettile, le prestazioni di velocità richieste al sistema di conteggio sono più elevate. Tuttavia, dato che un proiettile può impiegare alcuni millisecondi per percorrere la distanza misurata, anche questo tipo di musurazione è compresa nella capacità operativa di un sistema elettronico di conteggio relativamente semplice.

Si può far fronte alle applicazioni qui sopra riportate per mezzo di circuiti noti come **contatori seriali** (ripple-through counters).

Tuttavia, quando installiamo un calcolatore digitale per gestire per esempio la contabilità e il magazzino di una grande organizzazione, è necessario poter eseguire milioni di operazioni al secondo. Per far questo si devono impiegare circuiti noti come **contatori paralleli o contatori sincroni**.

I contatori sincroni possono essere utilizzati sia per basse che per alte velocità; la circuiteria richiesta è generalmente più complessa di quella dei contatori seriali.

7.2 Un contatore seriale in codice binario puro

Un contatore seriale in codice binario puro consiste in una catena di flip-flop *J-K* collegati nel modo di "commutazione" T mode (vedi Tabella 5.3), cioè essi devono avere le entrate *J* e *K* collegate al segnale logico "1" come mostrato dalla Fig. 7.2.
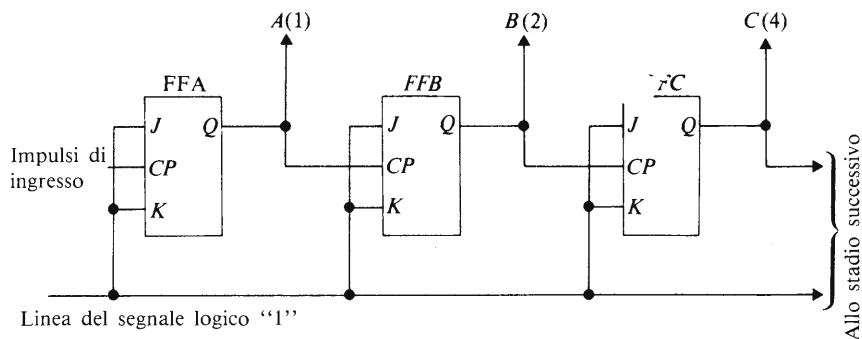
Fig. 7.2 Un contatore seriale che divide per otto

Il lettore ricorderà che se un flip flop *J-K* è collegato in questo modo, ogni qual volta il segnale di clock passa da "1" a "0", anche lo stato dell'uscita *Q* si modifica (vedi il paragrafo 5.7). Questo fatto sta alla base del contatore della Fig. 7.2.

Supponiamo che inizialmente tutte le uscite del contatore siano "0" ($A=0$, $B=0$, $C=0$). Questa situazione corrisponde a un tempo precedente l'istante t_1 nella Fig. 7.3; in essa sono rappresentate le forme d'onda alle varie uscite quando gli impulsi vengono applicati alla linea di ingresso.

Il cambiamento da "0" a "1" nell'istante T_1 dell'impulso di ingresso
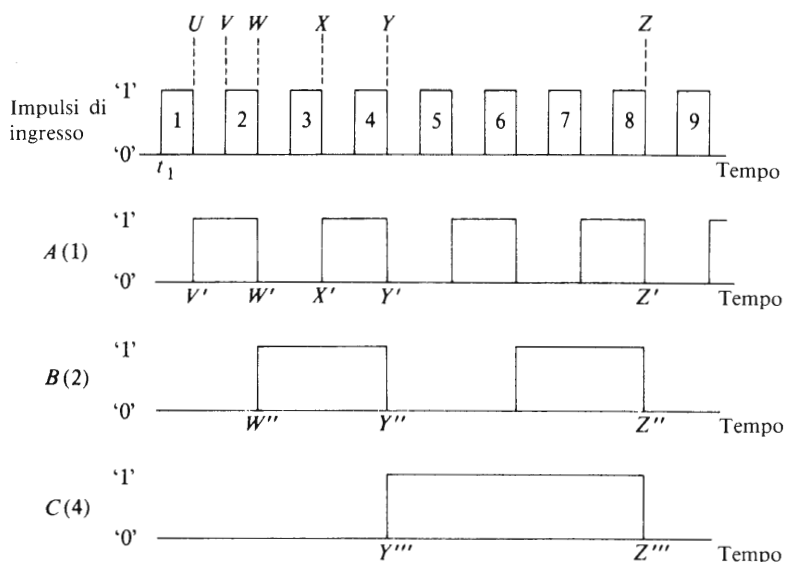
Fig. 7.3 Forme d'onda in un contatore seriale che divide per otto

numero 1 non ha alcun effetto sullo stato dell'uscita A. Tuttavia il cambiamento da "1" a "0" all'istante U dello stesso impulso provoca il cambiamento di A dallo stato "0" allo stato "1" (vedi il paragrafo precedente). Dopo un impulso d'ingresso completo gli stati delle uscite sono $A = 1$, $B = 0$, $C = 0$; questo corrisponde alla condizione riportata nella prima linea della Tabella 7.1.

L'uscita A è utilizzata come ingresso di clock per il flip-flop B (FFB); dato che il cambiamento dell'uscita A nell'istante di tempo U avviene tra lo stato "0" e lo stato "1", esso non ha alcun effetto sull'uscita del flip-flop B, cioè, B rimane a "0".

Quando l'impulso di ingresso numero 2 arriva all'ingresso di clock del flip-flop A, il cambiamento da "0" a "1" non ha alcun effetto sull'uscita A (che rimane nello stato di "1" logico). Tuttavia, il cambiamento da "1" a "0" all'istante W dell'impulso 2 provoca ancora un cambiamento nell'uscita A, questa volta da "1" a "0" (vedi W' nella Fig. 7.3). Dato che questo cambiamento è applicato all'ingresso di clock del flip-flop B esso provoca il cambiamento dell'uscita B da "0" a "1" (vedi W'' nella Fig. 7.3). L'uscita B è utilizzata come segnale di clock di ingresso per il flip-flop C, ma dato che il cambiamento avvenuto in W'' in B è un cambiamento da "0" a "1", esso non ha alcun effetto

Tabella 7.1 Uscite da un contatore divisore per otto

Numero di impulsi	*Uscite*			
	C (4)	B (2)	A (1)	
Condizione iniziale	0	0	0 ←	
1	0	0	1	
2	0	1	0	
3	0	1	1	
4	1	0	0	Si ripete
5	1	0	1	
6	1	1	0	
7	1	1	1	
8	0	0	0	
9	0	0	1	

sull'uscita C. Le uscite sono ora, perciò, $A=0$, $B=1$, $C=0$; situazione che corrisponde alla seconda riga della Tabella 7.1.

Il cambiamento da "1" a "0" nell'istante X (impulso di ingresso numero 3) provoca ancora una volta il cambiamento dell'uscita A da "0" a "1" (vedi X' nella Fig. 7.3; questo non ha alcun effetto nel flip-flop B, cosicchè si ha $A=1$, $B=1$, $C=0$ dopo il completamento dell'impulso 3 (vedi anche la riga 3 della Tabella 7.1). All'istante Y della Fig. 7.3, il cambiamento da "1" a "0" dell'impulso 4 provoca il cambiamento dell'uscita A da "1" a "0" (vedi Y' nella Fig. 7.3); questo cambiamento inoltre provoca un cambiamento nell'uscita B da "1" a "0" (vedi Y'').

Quest'ultimo cambiamento provoca infine il cambiamento dell'uscita C da "0" a "1" (*vedi Y'''*); le uscite sono ora $A=0$, $B=0$, $C=1$ (vedi anche la riga 4 della Tabella 7.1). Si lascia come esercizio ai lettori di controllare che questo processo continui fino a che, dopo il completamento del settimo impulso, le uscite sono $A=1$, $B=1$, $C=1$.

Il cambiamento dall'"1" allo "0" dell'impulso 8 (vedi Z nella Fig. 7.3) provoca il cambiamento di A dall'"1" allo "0" (vedi Z'), il cambiamento di B dall'"1" allo "0" (vedi Z'') e il cambiamento di C dall'"1" allo "0" (vedi Z''') cosicché $A=0$, $B=0$, $C=0$. I cambiamenti nelle uscite "scivolano in serie" o in sequenza ordinata attraverso il contatore (questo fatto giustifica il suo nome di 'contatore seriale'). Da que-

sto punto in poi la sequenza di conteggio viene ripetuta. Il lettore noterà dalla Tabella 7.1 che il contatore della Fig. 7.2 genera un codice binario puro da zero (a otto).

Uno svantaggio di questo tipo di contatore per quello che riguarda la sua velocità di funzionamento, è che ogni flip-flop impiega un po' di tempo per rilevare che l'uscita dallo stadio precedente è cambiata da "1" a "0". Qualora venisse utilizzata una lunga catena di flip-flop, bisogna dare sufficiente tempo a tutti i cambiamenti per propagarsi attraverso l'intera lunghezza del contatore prima che si possa "leggere" lo stato del contatore. Per questa ragione nei sistemi di conteggio ad alta velocità i contatori sincroni vengono preferiti ai contatori seriali.

Nei contatori sincroni gli impulsi di clock sono forniti simultaneamente a ciascun flip-flop; porte logiche addizionali collegano opportunamente le uscite dei singoli flip-flop in modo da fornire agli ingressi J e K degli stessi gli stati necessari alla commutazione del contatore nel codice desiderato.
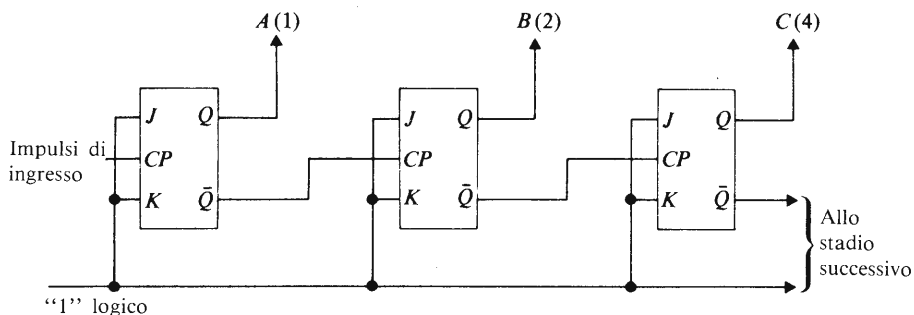
Fig. 7.4 Un contatore seriale all'indietro in codice binario puro

7.3 Un contatore seriale all'"indietro" in codice binario puro

Nella Fig. 7.4 si ha un contatore binario puro che conta all'"indietro" da un numero elevato a zero. Il lettore noterà che l'unica differenza tra la Fig. 7.2 e la Fig. 7.4 è che l'impulso di clock per il flip-flop B e per il flip-flop C è derivato dall'uscita $\overline{Q}$ precedente. La successione delle uscite del contatore è elencata nella Tabella 7.2. Si lascia come esercizio al lettore la spiegazione del funzionamento del circuito.

I circuiti di conteggio in avanti o all'indietro possono essere impiegati in determinate applicazioni. Per esempio, una macchina che deve impacchettare venti lattine in una scatola può utilizzare un contatore che conti

Tabella 7.2 Uscita da un contatore all'indietro in codice binario puro

Numero di impulsi	*Uscite*			
	C (4)	B (2)	A (1)	
Condizione iniziale	0	0	0 ←	
1	1	1	1	
2	1	1	0	
3	1	0	1	
4	1	0	0	Si ripete
5	0	1	1	
6	0	1	0	
7	0	0	1	
8	0	0	0	
9	1	1	1	

da zero a venti crescendo per indicare il numero di lattine nella scatola, o può utilizzare un contatore che conti da venti a zero per indicare il numero di spazi che rimangono nella scatola.

7.4 Un contatore seriale reversibile in codice binario puro

Abbiamo visto nei paragrafi 7.2 e 7.3 che un contatore che conti in avanti è realizzato connettendo l'uscita Q di uno stadio all'ingresso di clock del successivo e che è possibile realizzare un contatore che conti all'indietro quando si colloca l'uscita $\overline{Q}$ di uno stadio all'impulso di clock del successivo. Un **contatore** seriale **reversibile** che può contare sia in "avanti" che all'"indietro" è realizzato modificando l'accoppiamento tra i vari stadi come è riportato nella Fig. 7.5.

Ciascuna sezione di collocamento tra i vari stadi consiste di due porte AND e di una porta OR. Il segnale logico sulla linea di CONTROLLO DEGLI INGRESSI seleziona il canale attraverso cui avviene l'accoppiamento fra le diverse sezioni del contatore. Un segnale logico "1" applicato alla LINEA DI CONTROLLO apre infatti le porte G1, G2, G3, cosicché l'uscita Q di uno stadio è collegata con l'ingresso di clock dello stadio successivo. In questo modo il contatore conta in "avanti" quando l'INGRESSO DI CONTROLLO è "1". Nello stesso tempo viene applicato un segnale "0" alle porte G4, G5, G6 evitando così il funzionamento all'"indietro".

Fig. 7.5 Un contatore seriale bidirezionale (reversibile) in codice binario puro

Quando un segnale logico "0" viene applicato alla LINEA DI CONTROLLO, la porta G4 riceve all'ingresso un "1" logico. Questo segnale apre le porte G4, G5, G6 cosicché l'uscita *Q* di uno stadio è collegata all'impulso di clock dello stadio successivo. Questo meccanismo rende capace il contatore di funzionare all'"indietro". Nello stesso tempo, lo "0" sulla LINEA DI CONTROLLO impedisce il funzionamento delle porte G1, G2 e G3.

7.5 Un contatore seriale in codice 8421 BCD

La Fig. 7.6 rappresenta un contatore seriale che funziona secondo il codice BCD 8421: il codice generato dal contatore è riportato nella Tabella 7.3 (vedi anche paragrafo 6.4).

In aggiunta ai quattro flip-flop *J-K*, servono tre porte (G1, G2, G3) per fornire il collegamento corretto tra i vari stadi. Gli stadi operativi di queste porte sono riportati nella Tabella 7.3, nella quale una porta è "aperta" quando le informazioni possono fluire attraverso di essa, ed è "chiusa" quando impedisce il flusso di informazione.

Nel circuito riportato, tutti i flip-flop hanno i loro ingressi *J* e *K* colle-

Tabella 7.3 Sequenza di eventi della Fig. 7.6

Numero dell'impulso	*Stato delle uscite*				*Stato delle porte*	
	D (8)	C (4)	B (2)	A (1)	G1	G2
Condizioni iniziali	0	0	0	0	↑	↑
1	0	0	0	1		
2	0	0	1	0		
3	0	0	1	1		
4	0	1	0	0	aperta	chiusa
5	0	1	0	1		
6	0	1	1	0		
7	0	1	1	1	↓	↓
8	1	0	0	0	chiusa	aperta
9	1	0	0	1		
10	0	0	0	0	↑ aperta ↓	↑ chiusa ↓
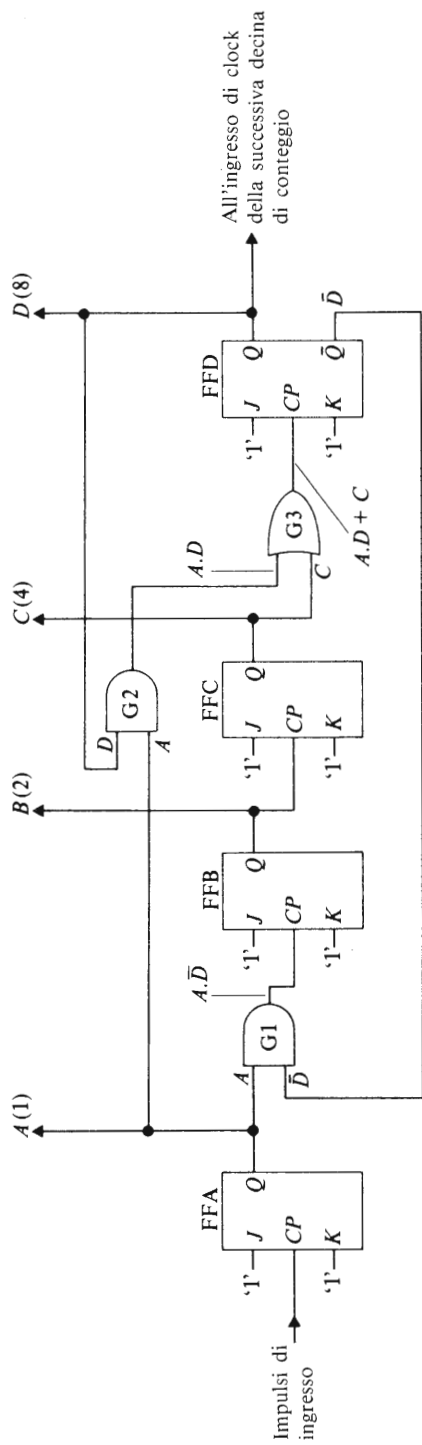
Fig. 7.6 Un contatore seriale BCD 8421

gati a segnali logici "1", cosicché essi funzionano nel modo T (vedi anche il paragrafo 5.7).

Assumiamo inizialmente che $A=B=C=D=0$; l'uscita $\overline{D}$ è perciò l'"1" logico. Durante l'intervallo di tempo in cui vengono applicati i primi sette impulsi di ingresso, il segnale logico "1" all'uscita $\overline{D}$ assicura che G1 sia "aperto" (cosicché il flip-flop A, il flip-flop B e il flip-flop C sono collegati in modo tale da funzionare come un contatore in "avanti").

Nello stesso tempo l'uscita D è uguale a 0, cosicché la porta G2 è "chiusa" e la sua uscita rimane nello stato di "0" logico.

Il cambiamento da "1" a "0" dell'ottavo impulso fa sì che le uscite A, B e C cambino da "1" a "0"; il cambiamento da "1" a "0" all'uscita C è trasmesso all'ingresso di clock del flip-flop D, e provoca il cambiamento dell'uscita D da "0" a "1" (cioè il cambiamento di $\overline{D}$ da "1" a "0"). L'effetto globale è che ora la porta G1 è chiusa e la porta G2 è aperta; fintanto che la porta G1 rimane chiusa, non possono venire trasmessi ulteriori impulsi al flip-flop B e al flip-flop C, le cui uscite rimangono a zero.

Il cambiamento da "1" a "0" del nono impulso di clock provoca il cambiamento dell'uscita A da "0" a "1", che viene trasmesso attraverso la porta G2 (ora aperta) e la porta G3 all'ingresso di clock del flip-flop D. Tuttavia, questo cambiamento da "0" a "1" non ha alcun effetto sullo stato del flip-flop D e le uscite sono $A=1$, $B=0$, $C=0$, $D=1$ (cioè il numero decimale nove). Quando il cambiamento dall'"1" allo "0" del decimo impulso di clock è applicato al flip-flop A, l'uscita A cambia dallo stato "1" allo stato "0". Questo cambiamento è trasmesso attraverso le porte G2 e G3 all'ingresso di clock del flip-flop D, che provoca il cambiamento dell'uscita D da "1" a "0". In tal modo, dopo dieci impulsi completi di ingresso, tutte le uscite sono di nuovo a zero.

In un contatore a molte decine l'uscita dello stadio D è applicata all'ingresso di clock del primo stadio del gruppo delle decine appena successive, formato da altri quattro flip-flop. Questo gruppo conta "decine" di impulsi. Dopo questo, il gruppo appena successivo conta le "centinaia" di impulsi.

7.6 Circuiti di conteggio integrati

Esistono molte versioni di contatori binari sotto forma di circuiti integrati. Un contatore comunemente utilizzato è il TTL 7490N (che corrisponde al FJJ141), riportato nella figura 7.7(a). Questo contatore fornisce la possibilità di ottenere un funzionamento in entrambe le modalità

(a)

(b) Collegamenti BCD

(c) Modalità di funzionamento "simmetrico"
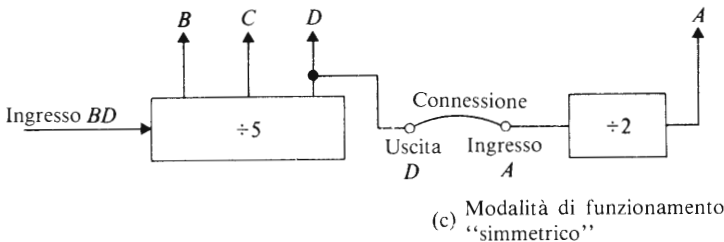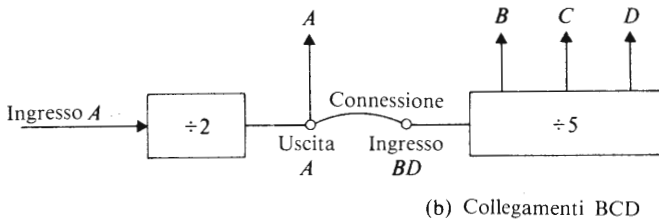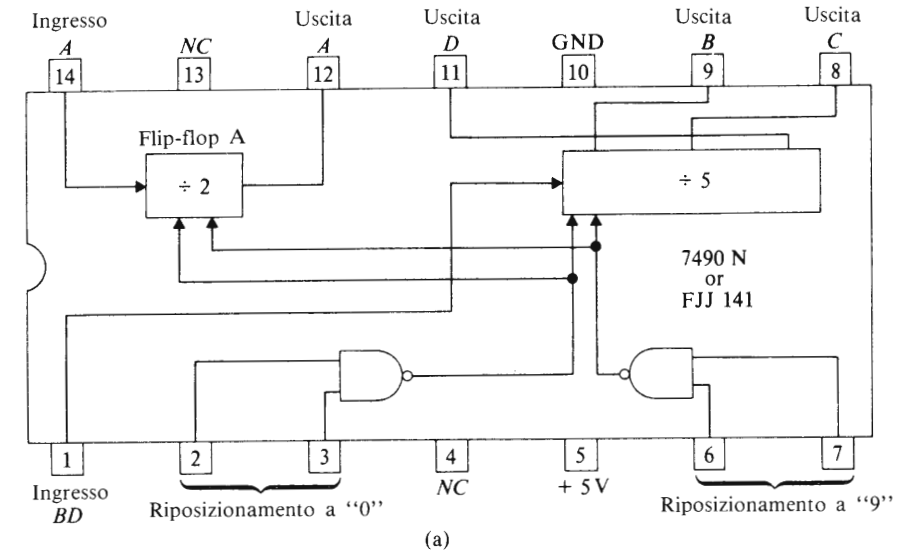
Fig. 7.7 (a) Un contatore a circuito integrato, (b) connessioni per il funzionamento nella modalità BCD 8421, (c) connessioni per il funzionamento come contatore "simmetrico" (vedi testo). I terminali contrassegnati da NC non sono collegati, mentre il terminale contrassegnato da GND è collegato al potenziale zero volt.

di operazioni "dividi per due" e "dividi per dieci", unita alla possibilità di funzionare come un contatore BCD 8421.

Quando il circuito viene utilizzato come contatore BCD 8421 (vedi Fig. 7.7b), l'ingresso *BD* deve essere collegato esternamente all'uscita *A*; l'ingresso *A* riceve gli impulsi in arrivo.

La modalità di funzionamento dividi-per-due è utilizzabile quando il flip-flop A è usato da solo, e la modalità di funzionamento dividi-per-cinque è utilizzabile quando gli impulsi di ingresso vengono applicati all'ingresso *BD* e le uscite sono prelevate da *B, C* e *D*. Il circuito integrato può essere anche utilizzato per fornire un contatore dividi-per-dieci in un codice detto "simmetrico" (vedi Tabella 7.4) collegando esternamente l'uscita *D* all'ingresso *A* (vedi Fig. 7.7c). Gli impulsi di ingresso vengono quindi applicati all'ingresso *BD*, e l'uscita è prelevata da *A*.

Vengono inoltre fornite al 7490N linee di riposizionamento diretto per mezzo di porte logiche, sia per posizionare a zero tutte le uscite, sia per produrre un codice *BCD* che conti in base nove decimale. Le tabelle della verità per le modalità di funzionamento *DCD* e SIMMETRICO sono riportate nella Tabella 7.4.

Tabella 7.4

Impulso di ingresso	*Modalità di funzionamento BCD ABCD*	*Modalità di funzionamento simmetrico BCDA*
0	0000	0000
1	1000	1000
2	0100	0100
3	1100	1100
4	0010	0010
5	1010	0001
6	0110	1001
7	1110	0101
8	0001	1101
9	1001	0011

7.7 Convertitori di codici

I sistemi logici utilizzano una tra le varie forme di codificazione dei dati in sequenze binarie, e l'uomo comunica informazioni con il codice decimale o per mezzo di parole che sono una forma di codice alfabetico.

E' necessario perciò utilizzare convertitori di codici per poter comunicare con i sistemi logici e viceversa.

Vogliamo considerare qui la progettazione di due tipi importanti di convertitori di codici, il primo dal sistema decimale al sistema binario (1); il secondo dal sistema binario al sistema decimale (2).

Progetto di un convertitore dal codice decimale a quello binario

La base del sistema è riportata nella Figura 7.8. I dieci segnali di ingresso sono ottenuti da una tastiera con dieci tasti che corrispondono alle cifre dallo 0 al 9. Quando premiamo un tasto viene applicato un "1" ad una linea di ingresso soltanto, mentre tutte le altre linee di ingresso hanno uno "0" logico. Per semplicità riportiamo lo schema per due soli tasti.

Il convertitore dal sistema decimale al sistema binario contiene la struttura logica che fornisce i segnali d'uscita a una o più delle linee di uscita *A, B, C, D.* I raggruppamenti di cifre codificati in uscita per ciascun tasto vanno elencati nella Tabella 7.5. Vediamo in questo modo che, quando viene premuto il tasto corrispondente al sei decimale, otteniamo degli "0" sulle linee di uscita *A* e *D* e degli "1" sulle linee *B* e *C.*

Per progettare la rete bisogna rilevare quale tasto decimale va premuto per fornire una certa uscita su una linea particolare. Per esempio, ve-

Tabella 7.5

Valore decimale (ingressi)	*Valore binario (uscite)*			
	A (8)	*B* (4)	*C* (2)	*D* (1)
0_{10}	0	0	0	0
1_{10}	0	0	0	1
2_{10}	0	0	1	0
3_{10}	0	0	1	1
4_{10}	0	1	0	0
5_{10}	0	1	0	1
6_{10}	0	1	1	0
7_{10}	0	1	1	1
8_{10}	1	0	0	0
9_{10}	1	0	0	1
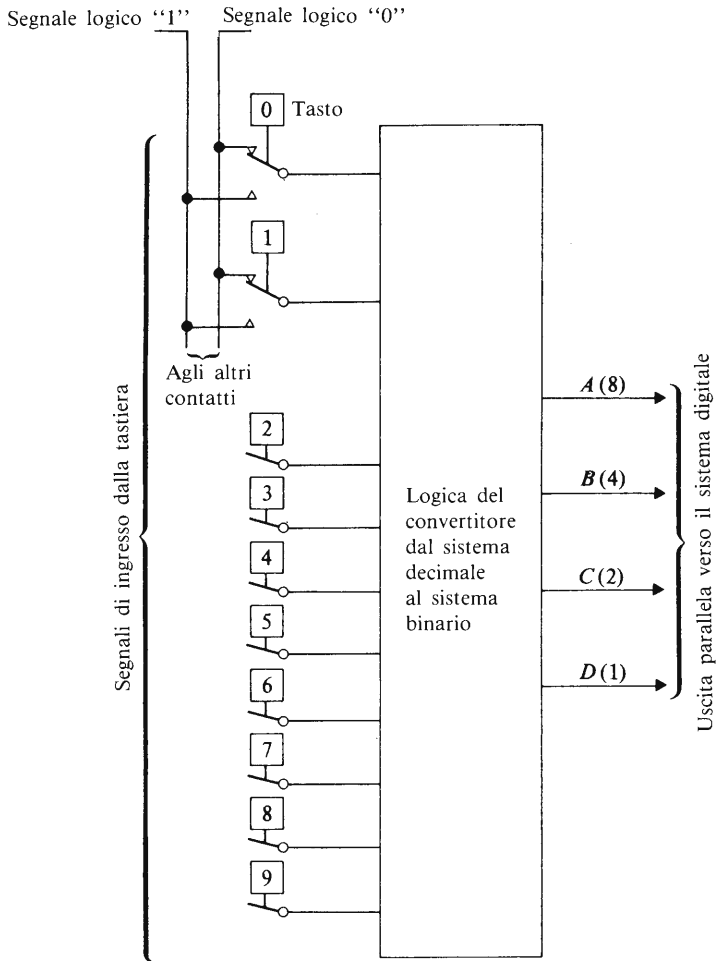
Fig. 7.8 - La struttura fondamentale di un convertitore dal sistema decimale al sistema binario

diamo nella Tabella 7.5, che perché appaia un "1" sulla linea A dobbiamo premere il tasto decimale 8 oppure (OR) il tasto decimale 9. L'espressione logica per l'uscita A potrebbe quindi essere scritta come

$$A = 8_{10} + 9_{10}$$
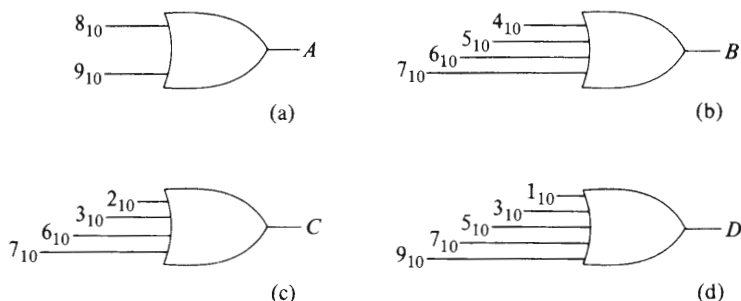
Fig. 7.9 Una forma di logica di transcodifica decimale-binario

Il diagramma a blocchi logici per la parte del circuito che ha a che fare con l'uscita A è riportato nella Fig. 7.9 (a) con i segnali 8_{10} e 9_{10} che provengono dai rispettivi tasti.

Dalla colonna per l'uscita B nella Tabella 7.5, vediamo che perché appaia un "1" su questa linea dobbiamo premere il tasto 4_{10}, oppure (OR) il tasto 5_{10}, oppure (OR) il tasto 6_{10}, oppure (OR) il tasto 7_{10}. Quindi

$$B = 4_{10} + 5_{10} + 6_{10} + 7_{10}$$

Il lettore potrà osservare che

$$C = 2_{10} + 3_{10} + 6_{10} + 7_{10}$$
$$D = 1_{10} + 3_{10} + 5_{10} + 9_{10}$$

La circuiteria logica per le uscite A, B, C e D è riportata nei diagrammi (a), (b), (c) e (d) rispettivamente della Fig. 7.9.

Progetto di un convertitore dal sistema binario BCD al sistema decimale

Quando un sistema digitale ha completato un calcolo, il risultato viene convertito in cifra decimale per mezzo di un convertitore dal sistema binario a quello decimale in modo che esso possa venire visualizzato su un indicatore. La struttura fondamentale di un convertitore dal sistema binario a quello decimale è riportata nella Fig. 7.10. Il sitema convertitore ha quattro linee di ingresso che portano "0" e "1" corrispondenti al valore binario che deve venir convertito, e ha dieci linee di uscita. Solo una alla volta tra le linee di uscita può portare un "1" logico; la ragione
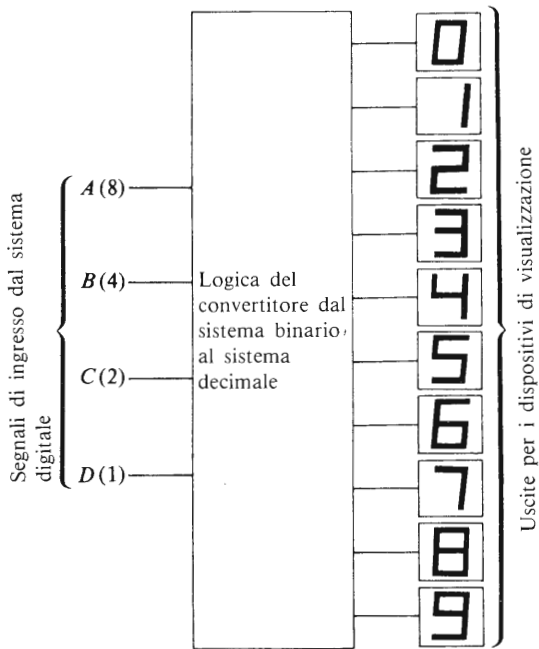
Fig. 7.10 Struttura fondamentale di un convertitore dal sistema binario al sistema decimale

di questo è che noi dobbiamo illuminare solo una cifra del visualizzatore alla volta.

La sequenza codificata è riportata nella Tabella 7.6, dove gli ingressi sono *A, B, C, D*. Guardando *lungo* la prima riga della tabella della verità (riga 0_{10}) vediamo che il valore decimale zero viene visualizzato quando $A=0$ e (AND) $B=0$ e (AND) $C=0$ e (AND) $D=0$ simultaneamente. Sotto queste condizioni deve apparire quindi un "1" logico sulla linea di uscita dello 0_{10} e deve apparire uno "0" logico su tutte le altre linee. In questo caso viene illuminato solo l'indicatore "zero". Possiamo scrivere l'espressione logica per l'indicatore numerico "zero" come

$$0_{10}=\overline{A}.\overline{B}.\overline{C}.\overline{D}$$

Il diagramma a blocchi che produce questa funzione è riportato nella Fig. 7.11(a). Perché venga visualizzato il valore decimale corrispondente all'unità, gli stati degli ingressi devono essere $A=0$ e (AND) $B=0$ e

(AND) $C=0$ e (AND) $D=1$ (vedi la seconda riga della Tabella 7.6). Per questo

$$1_{10}=\overline{A}.\overline{B}.\overline{C}.D$$

Il blocco che corrisponde a questa funzione è riportato nella Fig. 7.11(b). Applichiamo questa tecnica alle rimanenti uscite e si ottengono le altre porte della Fig. 7.11.

In alcuni casi è possibile ridurre o minimizzare il numero di linee di ingresso richieste nei convertitori di codici.

Tabella 7.6

Valore binario (ingressi)				*Valore decimale (uscite)*
A (8)	B (4)	C (2)	D (1)	
0	0	0	0	0_{10}
0	0	0	1	1_{10}
0	0	1	0	2_{10}
0	0	1	1	3_{10}
0	1	0	0	4_{10}
0	1	0	1	5_{10}
0	1	1	0	6_{10}
0	1	1	1	7_{10}
1	0	0	0	8_{10}
1	0	0	1	9_{10}

Tabella 7.7 Requisiti logici minimi per un convertitore dal sistema binario BCD a quello decimale

Valore decimale	*Trascrizione logica*	*Valore decimale*	*Trascrizione logica*
0	$\overline{A}.\overline{B}.\overline{C}.\overline{D}$	5	$B.\overline{C}.D$
1	$\overline{A}.\overline{B}.\overline{C}.D$	6	$B.C.\overline{D}$
2	$\overline{B}.C.\overline{D}$	7	$B.C.D$
3	$\overline{B}.C.D$	8	$A.\overline{D}$
4	$B.\overline{C}.\overline{D}$	9	$A.D$
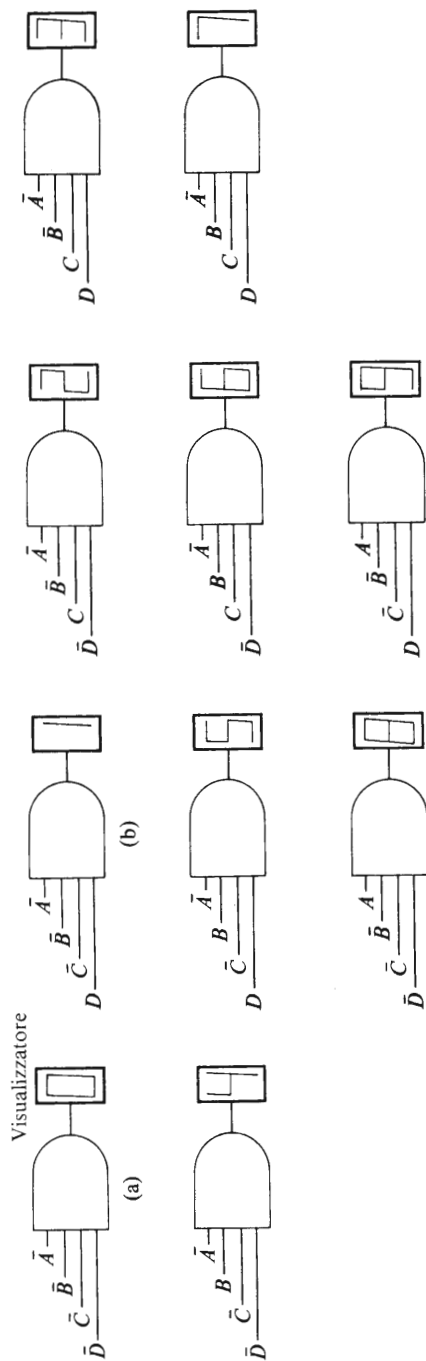
Fig. 7.11 Struttura logica fondamentale per un convertitore dal sistema binario BCD al sistema decimale

Un metodo per ottenere questo scopo è quello di analizzare attentamente la tabella della verità e di notare quali linee di ingresso non sono richieste; o, in alternativa, è posssibile usare tecniche speciali. I requisiti già minimizzati di un convertitore dal sistema binario al sistema decimale sono riportati nella Tabella 7.7; il lettore può provare a verificare egli stesso le soluzioni riportate nella tabella.

Ridurre il numero delle linee di ingresso richieste per alcune porte, permette di ridurre il numero delle potenziali sorgenti di guasti nel sistema e ne aumenta quindi la affidabilità.

7.8 Sequenza di controllo dei pistoni, 3 cicli

Alcuni schemi di controllo industriale richiedono un sistema di controllo che permetta di ripetere un ciclo di eventi molte volte.

Un esempio di questo tipo che noi considereremo è quello di un sistema che controlla due pistoni A e B che devono eseguire la sequenza A_+, B_+, A_-, B_- per tre volte dopo che viene premuto il pulsante di "marcia". La parte meccanica del sistema è riportata nella Fig. 7.12,
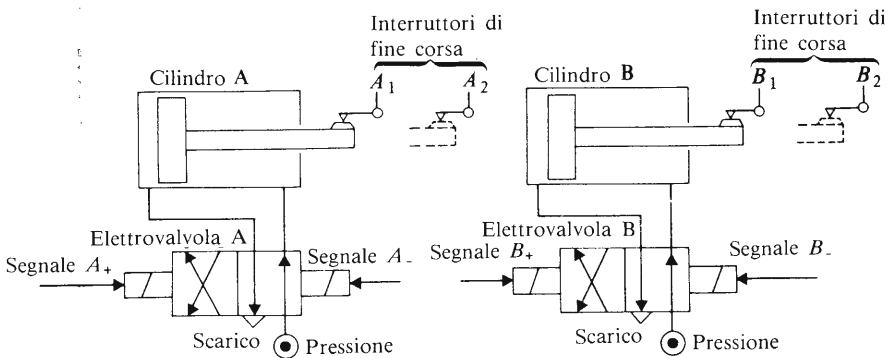
Fig. 7.12 I pistoni e i fine corsa nella sequenza di controllo del loro movimento

nella quale il movimento di ciascun pistone è controllato da un'elettrovalvola a due posizioni e quattro vie. Un possibile circuito di controllo è illustrato nella Fig. 7.13.

I due elementi *J-K* (il flip-flop *T* e il flip-flop *U*) formano un contatore che fornisce un controllo continuo sul numero di cicli già completati. La porta G2 è utilizzata per decodificare lo stato del contatore; infatti, l'uscita dalla porta G2 ($= T.U$) è zero ogni volta che T o U hanno valore logico "0" (situazione che si verifica nel primo e nel secondo ciclo),
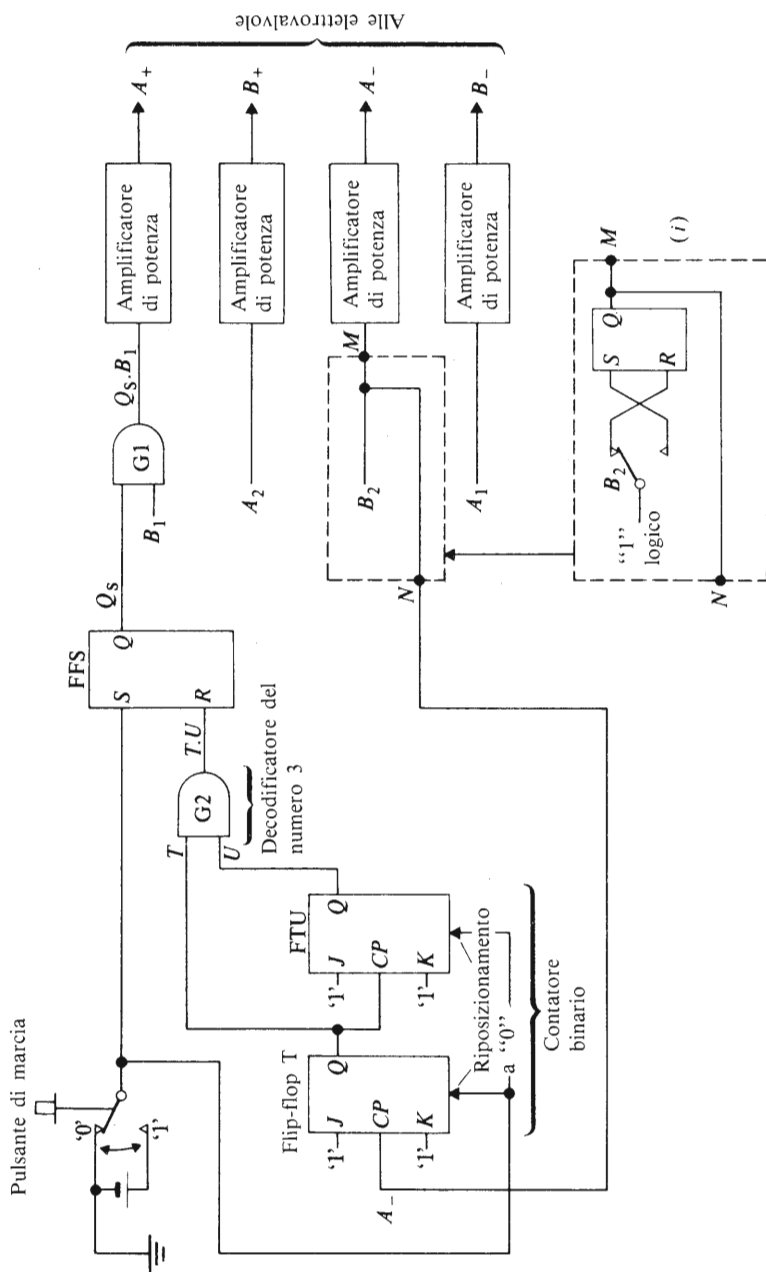
Fig. 7.13 Sequenza di controllo degli stantuffi: A_+, B_+, A_-, B_- tre cicli.

ed ha valore "1" quanto T e U sono "1" simultaneamente (situazione che si verifica verso la fine del terzo ciclo). Descriveremo ora il funzionamento del sistema.

I pistoni A e B sono completamente ritratti all'inizio del ciclo, cosicché le uscite degli interruttori di fine corsa A_1 e B_1 sono entrambe uguali ad "1". Premendo il pulsante di "marcia" l'uscita Q_s del flip-flop S viene posizionata (set) nella condizione di "1" logico, e questo rende uguale ad "1" l'uscita da G1. Inizia così la parte A_+ del ciclo. Nello stesso istante, il segnale di marcia fa sì che le uscite T e U del contatore vengano riportate (reset) a zero. Quando l'operatore rilascia il pulsante di marcia, il segnale sulla linea S del flip-flop S ritorna ad essere uno "0" logico, ma Q_s rimane un "1" logico dato che il flip-flop S non è stato riposizionato.

Al completamento dell'operazione A_+, l'uscita dall'interruttore A_2 diventa "1" e questo provoca l'inizio dell'operazione B_+. Quando l'operazione B_+ è completa, l'uscita dall'interruttore B_2 diventa un "1" logico, è questo provoca l'inizio dell'operazione A_-. La ragione per cui la sezione tra M e N è riportata in linea tratteggiata nella Fig. 7.13 associata all'interruzione B_2 verrà spiegata nel seguito.

Si nota che il segnale logico dell'interruttore B è utilizzato come ingresso di clock per il contatore; poiché l'operazione appena descritta provoca il cambiamento del segnale di clock da "0" a "1" essa non ha alcun effetto sullo stato dell'uscita del contatore.

Al completamento dell'operazione A_-, l'uscita dell'interruttore A_1 diventa "1" e questo provoca l'inizio dell'operazione B_-.

Appena il pistone B comincia a ritrarsi, il segnale dell'interruttore B_2 cade a zero; questa operazione fa sì che il valore immagazzinato nel contatore venga aumentato da zero a uno (si ricordi che B_2 è il segnale di clock). In questo modo il valore immagazzinato nel contatore è diventato uno prima che il primo ciclo fosse completato.

Alla fine del primo ciclo completo (che viene terminato dall'operazione B_-), l'uscita dell'interruttore B_1 diventa di nuovo un "1" logico. Questo segnale viene applicato all'ingresso della porta G1 insieme col segnale logico "1" proveniente dall'uscita Q del flip-flop S, e l'"1" logico risultante all'uscita della porta G1 dà inizio al secondo ciclo di operazioni dei pistoni.

Quando si compie per la seconda volta l'operazione B_- alla fine della seconda sequenza di operazioni, il valore immagazzinato nel contatore è aumentato da uno a due. Il completamento dell'operazione B_- permette alla terza operazione A_+ di cominciare (l'uscita della porta G1 è di nuovo "1").

Quando si verifica per la terza volta l'operazione B_-, il valore immagazzinato nel contatore è aumentato da due a tre; quando questo si verifica, le uscite del contatore sono $T=1$ e $U=1$ cosicché l'uscita della porta G2 diventa un "1" logico.

Questo segnale riposiziona (reset) l'uscita Q del flip-flop S sullo zero, inibendo in questo modo il funzionamento della porta G1. Il completamento della terza e ultima operazione B_- fa sì che l'uscita dell'interruttore B_1 torni ad essere un "1" logico, ma viene impedita la quarta operazione A_+ dallo "0" logico all'uscita Q del flip-flop S. Quando viene premuto ancora una volta il pulsante di marcia, il contatore viene riposizionato (reset) a zero, permettendo così che la sequenza si ripeta ancora tre volte.

Se l'interruttore B_2 è un interruttore di fine corsa, per superare problemi associati col "rimbalzo" di contatto dell'interruttore, sarà necessario sostituirlo con un circuito simile a quello riportato nella Fig. 7.13 nella zona indicata con (i). Se non si fa questa sostituzione, ogni "rimbalzo" dei contatti dell'interruttore B_2 provoca un aumento di una unità del valore immagazzinato nel contatore. Dato che si verificano molti impulsi di "rimbalzo" ogni volta che i contatti cambiano, il numero immagazzinato nel contatore non sarebbe una registrazione accurata del numero di cicli completati.

Capitolo 8

Registri di scorrimento (shift register)

8.1 Cos'è un registro di scorrimento?

Un "registro" è un gruppo di flip-flop interconnessi che immagazzina informazioni binarie. Un **registro di scorrimento** è un registro in cui i dati immagazzinati possono essere fatti scorrere attraverso il registro stesso sotto il controllo di una serie di impulsi di clock o impulsi di scorrimento.
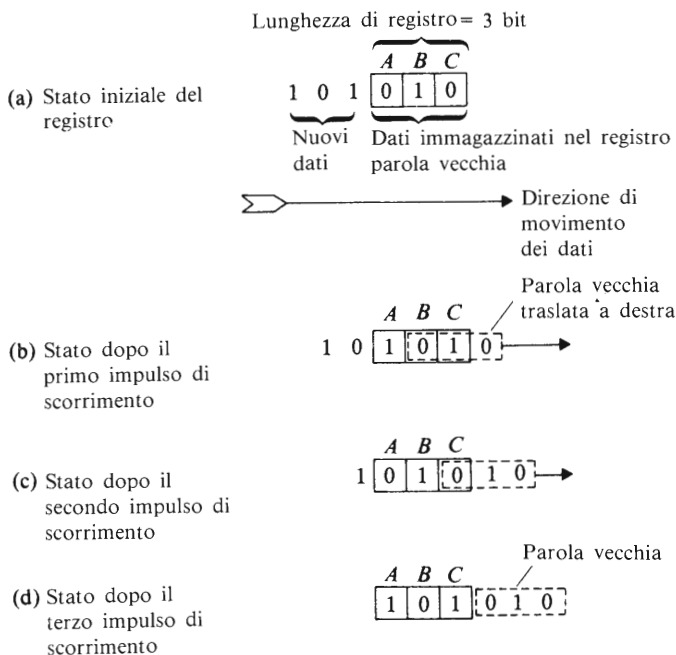
Fig. 8.1 La struttura fondamentale del funzionamento del registro di scorrimento

Il principio generale di funzionamento è rappresentato in Fig. 8.1. Supponiamo che il registro possa immagazzinare tre bit; in Fig. 8.1 (a) gli stati immagazzinati nei diversi stadi sono $A = 0$, $B = 1$, $C = 0$, e si vuole trasferire (traslare = to shift) la nuova parola binaria 101 nel registro.

Applichiamo ora una serie di impulsi di scorrimento (che sono in pratica una serie di impulsi applicati all'ingresso di clock di un flip-flop); ciascun impulso di scorrimento provocherà la traslazione a destra ("shift right") di una posizione dei dati immagazzinati (come questo si verifichi verrà descritto più avanti). In tal modo, dopo il primo impulso di scorrimento (vedi diagramma (b)), la parola vecchia è traslata di un posto a destra e il primo bit della parola "nuova" è traslato nell'elemento A del registro. Non ci dobbiamo preoccupare per ora di dove vada a finire la vecchia parola né da dove venga la parola "nuova".

Il secondo impulso di scorrimento provoca un'altra traslazione a destra, cosicché il registro contiene ora due bit della parola "nuova" (vedi il diagramma (c)). Infine dopo tre impulsi di scorrimento la parola vecchia è stata completamente sostituita dalla parola "nuova".

Dato che le informazioni vengono inserite un bit alla volta, diciamo che il registro ha un ingresso che funziona in **modo seriale**. In alternativa a questo, in alcuni registri è possibile far entrare le informazioni simultaneamente negli elementi del registro; questo modo di funzionare è detto a **ingresso parallelo**.

A seconda dal progetto del registro di scorrimento, è possibile lavorare in una fra le varie modalità qui elencate: ingresso seriale, uscita seriale (ilustrata in linea di principio in Fig. 8.1), oppure ingresso seriale, uscita parallela, oppure ingresso parallelo, uscita seriale.

8.2 Un registro di scorrimento che utilizza i flip-flop *J-K*

In Fig. 8.2 è riportato un tipo di registro di scorrimento a tre stadi. Questo tipo corrisponde al diagramma a blocchi della fig. 8.1 in quanto i dati vengono inseriti in modo seriale da sinistra e vengono traslati a destra per mezzo di impulsi di scorrimento (o di clock). La lunghezza del registro può essere ampliata connettendo Q, $\bar{Q}$ e i segnali di clock agli ingressi J, K e CP di altri elementi J-K.

Quando nei registri di scorrimento si usano elementi J-K il segnale applicato all'ingresso K deve essere sempre il complementare logico del segnale applicato all'ingresso J.

Per assicurarsi che questo avvenga, i dati che entrano in Fig. 8.2 sono
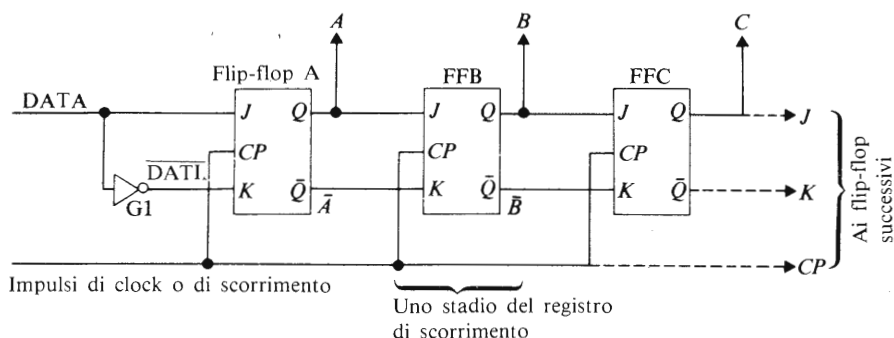
Fig. 8.2 Un registro di scorrimento a ingresso seriale

applicati all'ingresso K per mezzo di una porta NOT, G1; in questo modo, quando $J = 1$, $K = 0$ e viceversa. Il lettore ricorderà che quando l'impulso di clock applicato ad un flip-flop J-K viene cambiato dallo "0" logico all'"1", i dati presentati all'ingresso del flip-flop vengono trasferiti nello stadio "master"; quando l'impulso di clock cambia da "1" a "0" i dati immagazzinati nello stadio "master" vengono trasferiti allo stadio "slave".

Supponiamo ora che i dati inizialmente immagazzinati siano $A=0$, $B=1$, $C=0$ e che il segnale sulla linea dei DATI in arrivo sia l'"1" logico (vedi anche la situazione della Fig. 8.1a). Il cambiamento da "0" a "1" del primo impulso di clock provoca una traslazione nello stadio principale (master) del flip-flop A, del segnale "1" in arrivo nella linea dei DATI. Dato che l'uscita A è collegata all'ingresso J del flip-flop B e $\bar{A}$ al suo ingresso K, il passaggio da "0" a "1" del primo impulso di clock provoca il trasferimento dello stato dell'uscita A nello stadio "master" del flip-flop B. Utilizzando un ragionamento simile si può mostrare che lo stato di uscita B è traslato nello stadio "master" del flip-flop C.

Quando si verifica il cambiamento da "1" a "0" del primo impulso di clock, si provoca il trasferimento dei dati contenuti in ciascuno stadio "master" negli stadi "slave" dei flip-flop. In questo modo, dopo un impulso di clock completo, il segnale logico della linea DATI (un "1" in questo caso) compare all'uscita A; nello stesso tempo, il segnale che era all'uscita A (uno "0") appare all'uscita B, e il segnale che era all'uscita B (un "1") appare all'uscita C. Il segnale che era all'uscita del flip-flop C è stato traslato nello stadio successivo (nel caso che ve ne sia uno collegato), o viene perso.

Come risultato di un impulso di clock, quindi, la parola binaria immagazzinata nel registro è stata traslata di un posto a destra e il primo bit della nuova parola è stato trasferito nel registro (vedi anche Fig. 8.1b).

Ciascun successivo impulso di scorrimento provoca la traslazione di un posto a destra dei dati immagazzinati nel modo delineato dalla Fig. 8.1. Poiché tutti i dati si trovano simultaneamente sotto il controllo di un comune impulso di clock, il registro di scorrimento è considerato come un sistema sincrono.

8.3 Un registro di scorrimento reversibile

In alcune applicazioni è necessario poter trasferire i dati di un registro sia a destra che a sinistra. Abbiamo perciò bisogno di alcuni mezzi per "ruotare" i dati in entrambe le direzioni. Dall'analisi del paragrafo 8.2, ci si rende conto che i dati devono essere introdotti negli ingressi J e K del flip-flop e devono essere prelevati dalle uscite Q e $\overline{Q}$. Quando si fanno traslare i dati a destra (scorrimento "in avanti") nella Fig. 8.2, l'uscita del flip-flop A è applicata all'ingresso dei flip-flop B.

Quando abbiamo bisogno di traslare i dati a sinistra (scorrimento "all'indietro"), è necessario collegare l'uscita dello stadio C all'ingresso dello stadio B, e l'uscita dello stadio B, all'ingresso dello stadio A. L'uscita del registro è perciò prelevata in modo seriale dall'uscita dello stadio A (nello stesso modo in cui viene prelevata dallo stadio C quando si utilizza lo scorrimento a destra). In questo caso, ciascun impulso di clock provoca il trasferimento dei dati immagazzinati in C dallo stadio C allo stadio B e dei dati immagazzinati in B dallo stadio B allo stadio A.

Una realizzazione semplificata di una parte di una rete di interruttori per un registro di scorrimento reversibile è riportata in Fig. 8.3 (in pratica vengono sostituite porte logiche a tutti gli interruttori). Quando gli interruttori sono nella posizione mostrata, le connessioni sono identiche a quelle della Fig. 8.2 cosicché ciascun impulso di scorrimento provoca una traslazione a destra dei dati.

Le posizioni dei contatti degli interruttori sono controllate da un segnale elettronico; per traslare i dati a sinistra, questo segnale provoca il cambiamento di posizione di *tutti* gli interruttori. Esso fa sì che gli interruttori S1 e S2 colleghino l'uscita C all'ingresso J del flip-flop B; e che gli interruttori S1A e S2A colleghino l'uscita $\overline{C}$ all'ingresso K del flip-flop B. Questo significa che i dati sono traslati a "sinistra".
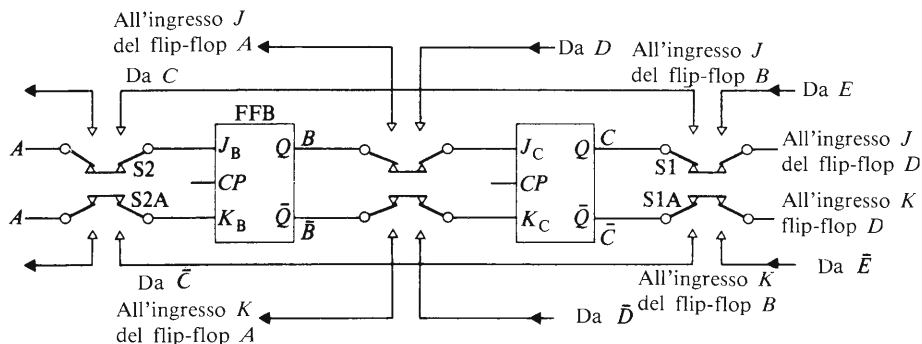
Fig. 8.3 Schema fondamentale di un registro di scorrimento reversibile

Il lettore interessato ai circuiti logici per il controllo dei registri di scorrimento reversibili troverà ulteriori dettagli in testi più specializzati .

8.4 Applicazioni industriali dei registri di scorrimento

Un'applicazione industriale del registro di scorrimento è quella del rigetto automatico di un componente difettoso da una linea di produzione automatizzata. Il movimento dei componenti lungo la linea di produzione può venir monitorato per mezzo di sensori e può essere indicato elettronicamente dal movimento di un "1" logico lungo un registro di scorrimento. Se un sensore indica un difetto nel componente, l'"1" logico può essere sostituito da uno "0" per mezzo della possibilità di riposizionamento (reset) o cancellazione del flip-flop *J-K* (vedi paragrafo 5.7). In questo modo gli "1" e gli "0" sul registro rappresentano rispettivamente i componenti "soddisfacenti" e quelli "difettosi". Quando un componente è trasferito da uno stadio operativo ad un altro, il sistema elettronico "guarda" allo stato del registro di scorrimento per "vedere" se il componente è difettoso. Se viene trovato uno "0" sul registro, vengono inibite ulteriori operazioni sul pezzo in questione. Ad un certo punto della sequenza di operazioni, il componente difettoso è automaticamente espulso da un circuito che riconosce lo "0" nel registro.

Un'altra applicazione consente la trasmissione di dati tra parti di una installazione che si trovano distanti le une dalle altre. Consideriamo il caso in cui una lontana stazione di pompaggio debba venire controllata da una stazione centrale. Si può usare un registro di scorrimento per controllare il funzionamento del sistema: i flip-flop del registro possono essere distribuiti tra punti o stazioni che si trovano a parecchi chilometri

di distanza; una parte del registro di scorrimento si trova alla stazione di pompaggio. Consideriamo il caso più semplice in cui il registro di scorrimento ha solo due parti (corrispondenti alle parti A e B della Fig. 8.2), con la parte A nella stazione di controllo principale e la parte B nella stazione ausiliaria.

Per iniziare il pompaggio, si fa entrare un "1" nello stadio A nella stazione di controllo principale. Dopo che un impulso di clock è stato completato (il che si verifica in una frazione di millisecondo), l'"1" appare alla stazione ausiliaria. Il sistema logico della stazione di pompaggio, nel momento in cui rileva la presenza dell'"1" logico, dà inizio a un avviso adatto per la preparazione alla partenza, seguito dalla partenza effettiva della pompa.

Chiaramente in una situazione pratica, il sistema sarebbe più complicato e includerebbe altri flip-flop nel registro per il controllo delle valvole, ecc. Inoltre un sistema pratico avrebbe bisogno di una qualche forma di registro di scorrimento reversibile dato che è necessario trasmettere informazioni di ritorno alla stazione di controllo centrale circa il flusso e il livello del fluido, ecc.

8.5 Un contatore ad anello (ring counter)

Un **contatore ad anello** è un registro di scorrimento a ingresso seriale e uscita seriale, la cui uscita finale Q è collegata con l'ingresso J e la cui uscita $\overline{Q}$ è collegata all'ingresso K. Inoltre, in ogni istante di tempo, il contatore immagazzina solo *un* "1" logico, mentre i restanti flip-flop immagazzinano "0".

La Fig. 8.4(a) rappresenta un contatore ad anello a tre stadi; supponiamo che inizialmente gli stati del registro siano: $A=1$, $B=0$, $C=0$; la sequenza di eventi per i primi quattro impulsi di clock è riportata in Tabella 8.1. Dopo un impulso di clock completo, l'"1" precedentemente immagazzinato nel flip-flop A è trasferito nel flip-flop B, lo "0" precedentemente immagazzinato nel flip-flop B è trasferito nel flip-flop C, e lo "0" precedentemente immagazzinato nel flip-flop C è trasferito nel flip-flop A. Si può dedurre quindi che una sequenza di impulsi di clock provoca la circolazione dell'"1" logico attorno al contatore ad anello come riportato nella Tabella 8.1.

Possiamo usare il dispositivo della Fig. 8.4(a) come un contatore dato che, quando compare un "1" in A, esso corrisponde alla condizione iniziale della sequenza, o al conteggio di zero; quando compare in B corrisponde al conteggio di una unità e, quando compare in C, corrisponde
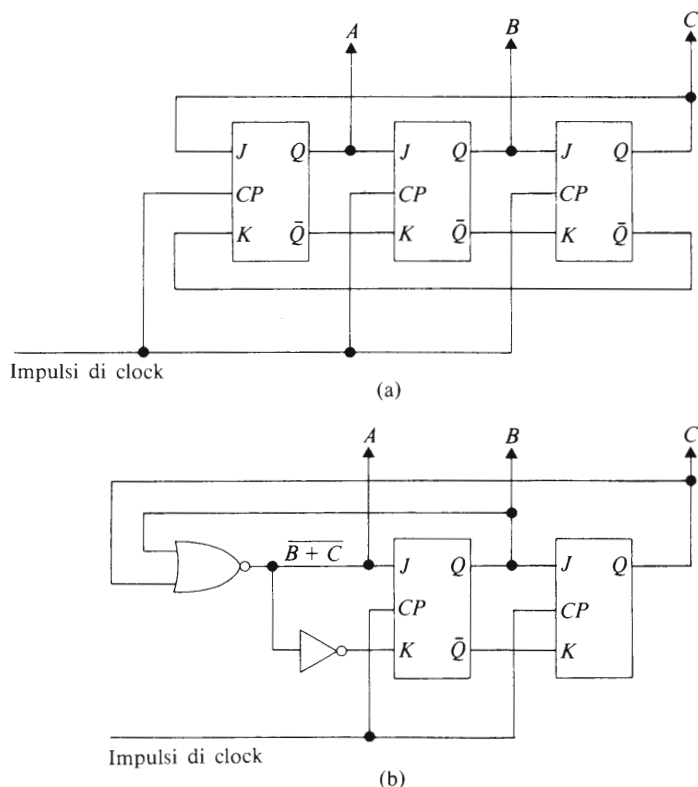
Fig. 8.4 Contatori ad anello

al conteggio di due unità. In questo modo, osservando la posizione dell'"1" nel registro, sappiamo quanti impulsi di clock sono stati applicati.

Se gli impulsi di clock sono generati dai componenti di una linea di produzione che passano davanti ad un sensore, "contiamo" effettivamente il numero dei componenti. Non c'è limite al numero dei flip-flop che possiamo usare nel contatore ad anello; se utilizziamo dieci elementi *J-K*, otteniamo un contatore decimale.

Come è già stato ricordato nei capitoli precedenti, una limitazione pratica di tutti i flip-flop è che al momento di accensione, si verifica una situazione di gara, cosicché non abbiamo alcuna conoscenza diretta di quale uscita sarà l'"1" logico. Così, all'istante di accensione, le uscite dal contatore ad anello potrebbero assumere un valore qualsiasi ed è

Tabella 8.1 Uscite da un contatore ad anello a tre stadi

Impulso di clock	*Uscite*		
	A	*B*	*C*
Condizioni iniziali	1	0	0 ←
1	0	1	0
2	0	0	1
3	1	0	0
4	0	1	0

molto probabile che non si verifichi nessuna delle combinazioni della Tabella 8.1. Cioè, il contatore inizialmente è caricato in maniera elettronicamente casuale: ne segue che qualsiasi cosa contenga inizialmente il contatore, abbiamo bisogno di un circuito che generi automaticamente una delle sequenze codificate nella Tabella 8.1; da qui in poi i collegamenti del contatore ad anello assicureranno che si verifichi la corretta sequenza di "0" e di "1". Il circuito della Fig. 8.4(b) assicura che questo accade dopo non più di un impulso di clock; il lettore noterà che questo circuito richiede l'utilizzazione di due flip-flop *J-K*. Si lascia come esercizio la verifica che il circuito della Fig. 8.4(b) genera in effetti la sequenza codificata nella Tabella 8.1.

Se abbiamo bisogno di costruire un contatore ad anello che usi il sistema decimale, utilizzando il tipo di circuito della Fig. 8.4(b) abbiamo bisogno di nove flip-flop, di una porta NOR a nove linee di ingresso e di una porta NOT. Quali che siano gli stati iniziali immagazzinati nel contatore all'istante di accensione, la sequenza operativa corretta si verificherà dopo non più di otto impulsi di clock.

Si può ora mettere alla prova le conoscenze acquisite fino a questo momento, progettando un circuito che utilizza un anello contatore per sostituire il contatore binario puro nella sequenza di controllo dei pistoni che completano tre cicli prima di fermarsi, come descritto nel paragrafo 7.8.

8.6 Trasmissione a divisione di tempo - Un'applicazione di un contatore ad anello

Il processo di invio di numerosi messaggi lungo uno stesso filo si dice "multiplessaggio", termine di gergo usato correntemente per indicare la selezione comandata tra diversi segnali.

In questo paragrafo consideriamo la struttura di un sistema digitale di acquisizione dei dati che permette di trasmettere tre segnali *W,X,Y* da tre sensori separati lungo un solo filo DATI. In pratica, il sistema potrebbe avere a che fare con molti sensori, ma a titolo esemplificativo limitiamo il nostro esame al caso semplice di tre sensori.

Il diagramma a blocchi di un semplice sistema di multiplessaggio a divisione di tempo (T.D.M. = time division multiplexing) è riportato in Fig. 8.5. I segnali dai sensori *W,X* e *Y* sono collegati rispettivamente agli ingressi delle porte AND G1, G2 e G3. Inoltre, collegate agli ingressi di queste porte, sono le uscite *Q* dei primi tre stadi di un contatore ad anello a quattro stadi (le uscite *A, B* e *C*, rispettivamente).
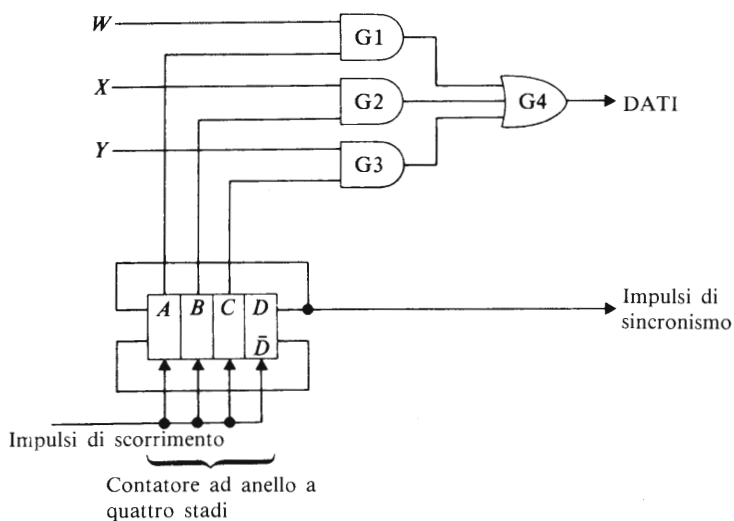
Fig. 8.5 La struttura fondamentale di unsistema TDM

Assumiamo che *A* sia inizialmente uguale a "1" e che *B, C* e *D* siano uguali a "0". In questo istante di tempo il segnale "1" dell'uscita *A* del contatore ad anello è applicato alla porta G1, "aprendo" così la porta alla trasmissione dello stato del sensore *W*. Lo stato di questo sensore (che può essere sia "1" che "0") è trasmesso attraverso la porta AND G1 alla porta OR G4 e da qui lungo la linea di trasporto dei DATI, fino al ricevitore.

Nello stesso tempo vengono applicati "0" logici dal contatore ad anello alle porte G2, G3, impedendo così sia il loro funzionamento, che la

trasmissione degli stati dei sensori X e Y.

Dopo il completamento del primo impulso di clock, l'"1" nell'anello contatore è trasferito all'uscita B (mentre $A=0$, $C=0$ e $D=0$). Questo apre la porta G2 e trasmette lo stato del sensore X attraverso la porta G4 lungo la linea di trasporto dei DATI. In questo intervallo di tempo, gli "0" applicati dalle uscite A e C del contatore ad anello inibiscono il funzionamento delle porte G1 e G3, impedendo così che gli stati dei sensori W e Y vengano trasmessi.

Dopo il secondo impulso di clock l'"1" nell'anello contatore è trasferito all'uscita C. Questa apre la porta AND G3 (a G1 e G2 viene ora impedito di funzionare) permettendo allo stato del sensore Y di venire trasmesso lungo la linea di trasporto dei DATI. In questo modo l'uscita di ciascun sensore è scandita o multiplessata in sequenza di tempo, e le informazioni vengono trasmesse in sequenza temporale lungo la linea di trasporto dei DATI.

Dopo il terzo impulso di clock l'"1" del contatore compare all'uscita D che provoca la trasmissione di un "1" logico lungo la linea di sincronismo o linea SYNC. Questo segnale informa il ricevitore che una scansione completa dei sensori è stata portata a termine e che un'altra scansione sta per cominciare.

In questo momento le uscite A, B, e C sono "0" e non viene perciò trasmessa alcuna informazione lungo la linea di trasporto dei DATI.

L'impulso di clock successivo trasferisce ancora una volta l'"1" all'uscita A, e la sequenza di scansione viene ricominciata.

8.7 Contatore a codice Johnson o "Twisted-Ring Counter"

Una limitazione fondamentale del contatore ad anello è che il massimo numero a cui può arrivare contando è limitato a un valore uguale al numero di flip-flop utilizzati (o al numero di flip-flop più uno come nel caso di Fig. 8.4b). Per esempio l'anello contatore a tre stadi di Fig. 8.4(a) può trattare solo numeri fino al due decimale, (cioè con 0, 1, e 2); un contatore decimale (0-9) richiede invece dieci flip-flop.

Utilizzando un "twisted ring counter" del tipo riportato in Fig. 8.6, possiamo generare una sequenza in codice la cui lunghezza è uguale al doppio del numero di flip-flop utilizzati.

In questo contatore l'ingresso J_A è alimentato da $\overline{C}$ e l'ingresso K_A è alimentato da C. Supponiamo che inizialmente sia $A=1$, $B=0$ e $C=0$. Quando viene applicata una serie di impulsi agli ingressi di clock dei flip-flop, la sequenza di uscite che si ottiene è uguale a quella riportata nella Tabella 8.2.
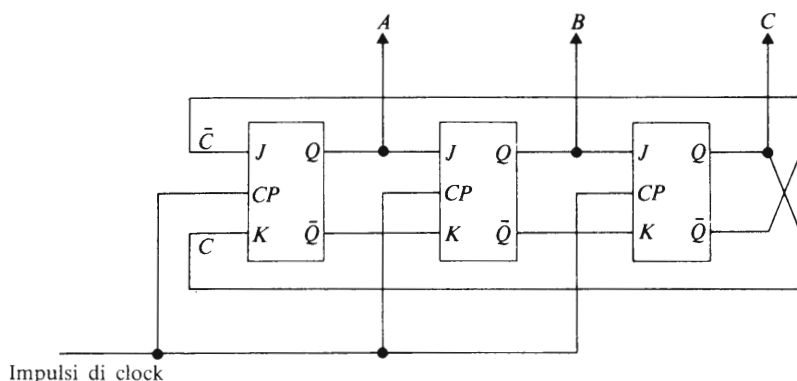
Fig. 8.6 Un contatore in codice Johnson ovvero un "twisted-ring counter"

Tabella 8.2 Uscite da un "twisted-ring counter" a tre stadi

Impulso di clock	*Uscite* A	B	C	
Condizione iniziale	1	0	0 ←	
1	1	1	0	
2	1	1	1	
3	0	1	1	Si ripete
4	0	0	1	il ciclo
5	0	0	0	
6	1	0	0	

Dopo il primo impulso di clock lo stato di $\bar{C}$ (=1) compare all'uscita A, e gli stati precedenti di A e B diventano rispettivamente i nuovi stati di B e C. In questo modo lo stato dell'uscita A è uguale allo stato di $\bar{C}$ prima dell'ultimo impulso di clock. Ciascun impulso di clock, quindi, trasla lo stato dello stadio A a destra. Dopo 5 impulsi di clock tutte le uscite sono zero, e l'impulso di clock successivo provoca la ripetizione del ciclo. Questo tipo di contatore è utilizzato in molti strumenti elettronici.

Il lettore può sviluppare per esercizio la sequenza di codice generata nel caso in cui all'istante di accensione le uscite siano $A=1$, $B=0$, $C=1$.

8.8 Un registro di scorrimento dinamico a MOSFET

Nel capitolo 5 (paragrafo 5.9) abbiamo accennato all'argomento degli elementi dinamici di memoria: in essi l'informazione è immagazzinata sotto forma di carica elettrica (o mancanza di carica elettrica) nel condensatore gate-canale di un MOSFET.

La struttura fondamentale di una forma di registro di scorrimento dinamico è riportata in Fig. 8.7. Uno stadio del registro di scorrimento consiste di due parti, ciascuna contenente una porta NOT MOSFET simile a quelle descritte nel paragrafo 3.7, collegate tra loro per mezzo di interruttori (in pratica gli interruttori sono sostituiti da dispositivi MOSFET).

Il condensatore C_1 della porta G1 e il condensatore C_2 della porta G2, ecc. rappresentano il condensatore gate-canale di un MOSFET. Il funzionamento degli interruttori di collegamento è controllato dai segnali sulle linee ϕ_1 e ϕ_2 che sono alimentate da un **clock** cosiddetto **a due fasi**.
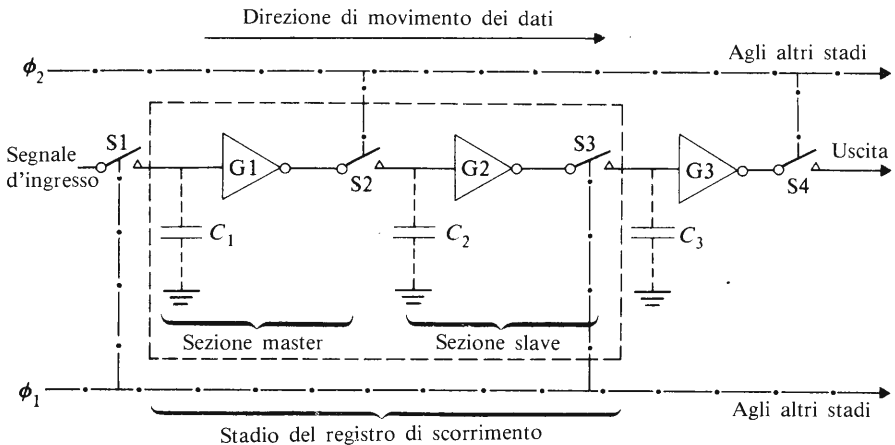
Fig. 8.7 Struttura fondamentale di un registro di scorrimento dinamico MOSFET

Esso consiste in una serie di impulsi concatenati in modo che la spaziatura temporale degli impulsi stessi sia tale per cui i due interruttori associati a uno stadio del registro di scorrimento (S2 e S3 per esempio) non siano mai chiusi simultaneamente. Il generatore di impulsi a due fasi utilizzato qui non deve venir confuso con un generatore elettrico a due fasi, dato che il primo consiste di due treni di impulsi rettangolari che variano tra lo "0" logico e l'"1" logico, mentre il secondo consiste in

due tensioni sinusoidali sfasate una rispetto all'altra.

Quando ϕ_1 è "1" logico, l'interruttore S1 si chiude per permettere che il dato di ingresso (un "1" o uno "0") venga trasferito al condensatore C_1. Se il segnale di ingresso è un "1" logico, allora l'uscita della porta G1 è uno "0" logico.

Nell'istante in cui $\phi_1 = 1$, il segnale sulla linea ϕ_2 è lo "0" logico, cosicché l'interruttore S2 è aperto. Nello stesso istante in cui l'interruttore S1 è chiuso, anche l'interruttore S3 è chiuso. Quest'ultimo permette al segnale in uscita dalla porta G2 di venir trasferito alla sezione "master" del secondo stadio.

Quando il segnale sulla linea ϕ_1 si riduce a "0" esso provoca l'apertura degli interruttori S1 e S3. Un "1" logico viene applicato perciò alla linea ϕ_2 e provoca la chiusura dell'interruttore S2; l'uscita dalla porta G1 corrispondente a uno "0" logico, scarica il condensatore C_2, facendo in modo che l'uscita della porta G2 sia un "1" logico. I dati vengono perciò trasferiti attraverso uno stadio del registro dopo un ciclo completo di ϕ_1 e ϕ_2.

Capitolo 9

Sistemi di memoria

9.1 Tipi di elementi di memoria

Gli elementi di memoria possono venir generalmente suddivisi in due famiglie, le memorie non volatili e le memorie volatili.

Le memorie non volatili mantengono l'informazione quasi a tempo indeterminato, anche nel caso di mancanza di tensione. Inclusi in questo gruppo sono i dispositivi a immagazzinamento magnetico, per esempio i nastri, i dischi e i nuclei magnetici. Soltanto un numero ristretto di circuiti a semiconduttori può fornire la non volatilità.

Le memorie volatili mantengono l'informazione fin tanto che ad esse viene fornita l'alimentazione. La maggioranza delle memorie a semiconduttori è di questo tipo.

I due sottogruppi delle memorie volatili sono le memorie statiche e le memorie dinamiche. Questi tipi di memorie sono già stati menzionati in precedenza nel testo, e vengono semplicemente qui ricordati al lettore. In una **memoria statica** l'informazione è mantenuta per mezzo dei cammini di reazione dei flip-flop *S-R* (vedi il paragrafo 5.2). Nelle **memorie dinamiche**, i dati vengono immagazzinati sotto forma di carica elettrica sulla capacità gate-canale di un MOSFET (vedi paragrafo 5.9 e 8.8); la carica immagazzinata in questo modo decade col tempo e ha bisogno di essere periodicamente "rinfrescata".

9.2 Memorie ad accesso casuale (RAM)

Le informazioni sotto forma di cifre binarie sono immagazzinate in una specifica locazione della memoria di un sistema digitale. Ciascuna locazione ha un suo **indirizzo** specifico, tanto quanto il lettore ha un suo indirizzo di residenza.

Se è necessario determinare il "contenuto" di una particolare locazione della memoria occorre l'indirizzo di quel punto; l'indirizzo di una cella è analogo al codice di avviamento postale.

In una **memoria ad accesso casuale** (RAM = random access memory) possiamo indirizzare una qualsiasi cella a caso. Questo significa che possiamo andare direttamente alla cella della memoria senza passare per nessun punto intermedio. Una memoria ad accesso casuale è talvolta descritta come una **memoria di lettura e scrittura** dato che le informazioni possono essere scritte nella cella o lette dalla cella, in un modo abbastanza analogo a quello con cui il lettore manda o riceve lettere al suo recapito.

L'organizzazione di una RAM a semiconduttore è riportata in Fig. 9.1. Ciascuna cella di memoria, per esempio, potrebbe essere un flip-flop, con le varie celle disposte in forma matriciale o rettangolare. L'indirizzo di una cella particolare è dato dal suo numero di riga e dal suo numero di colonna; in questo modo la cella M1, 2 si trova all'incrocio
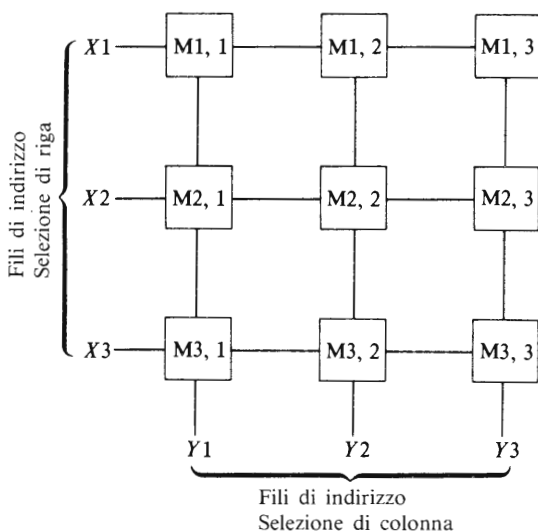
Fig. 9.1 Sistemazione matriciale di celle di una RAM a semiconduttore

della prima riga e della seconda colonna, la cella M2, 3 si trova all'incrocio della seconda riga e della terza colonna, ecc. Una cella è "indirizzata" attivando in modo opportuno la riga e le colonne appropriate; per cui, la cella M2, 1 è "indirizzata" attivando la linea $X2$ e la colonna $Y1$.

Avendo cosí selezionato una cella, possiamo scrivere nella cella o leggere dati dalla cella per mezzo di un filo (non riportato in figura), che

corre attraverso ogni cella della matrice. Questo procedimento di selezione della cella è noto come **selezione *X-Y* o selezione di coincidenza**.

In modo alternativo a questo, una parola binaria completa può venire letta dalla memoria per mezzo di un processo noto come **selezione di parola** o **selezione lineare**, nella maniera seguente: vengono attivati simultaneamente tutti fili di "selezione di riga" insieme ad un solo filo di "selezione di colonna", per esempio *Y*2. Utilizzando un filo "sense" addizionale per riga (nella terminologia inglese sense = senso, non riportato in figura), vengono indirizzati simultaneamente gli stati di M1,2 M2,2 e M3,2 sia per "scrivere" che per "leggere" dati.

Gli elementi di memoria nella RAM possono essere elementi statici o dinamici. In questo secondo caso, il sistema elettronico di controllo deve inserire periodicamente un breve intervallo di tempo per "rinfrescare" i contenuti immagazzinati nella RAM.

L'indirizzo di ciascun filo di riga e di ciascun filo di colonna è indicato per mezzo di un indirizzo binario, cosicché la cella nella prima riga e seconda colonna ha l'indirizzo binario 01,10 (corrispondente al numero binario 1 e al numero binario 2). Perciò, una parola di indirizzamento a due bit può indirizzare fino a quattro fili. Se utilizziamo una parola di indirizzamento di cinque bit, possiamo indirizzare fino a $2^5 = 32$ fili. Una matrice di 32×32 elementi di memoria contiene 1024 celle di memoria e viene indicata come avente una **capacità di immagazzinamento di 1K**. Perciò due parole di cinque bit possono venire utilizzate per indirizzare 1024 celle diverse in una RAM; una delle 2 parole di cinque bit verrà utilizzata per indirizare i fili *X* e l'altra parola per indirizzare i fili *Y*.

9.3 Memorie di sola lettura

Una memoria di sola lettura (ROM = read-only) contiene informazioni che sono "lette" frequentemente e che vengono modificate raramente oppure mai. I dati immagazzinati sono "scritti" nella memoria al momento della produzione o della messa in opera di una apparecchiatura particolare.

Una piccola sezione di una possibile forma di una ROM elettricamente programmabile è riportata in Fig. 9.2. In questo caso, ciascun elemento di memoria consiste in un diodo, in serie al quale è posto un fusibile a semiconduttore. Ciascun fusibile può essere "bruciato" al momento di fabbricazione indirizzando la cella specifica e applicando un impulso di corrente a essa. Quando la memoria deve ritenere uno stato

"1", per esempio, il fusibile è lasciato intatto; quando deve ritenere uno "0", la connessione viene fusa. Il tipo di ROM mostrato nella Fig. 9.2 è descritto come una **memoria programmabile di sola lettura** (PROM = Programmable Read-Only Memory) e non può essere riprogrammata. Queste memorie vengono anche descritte come PROM **a fusibili**.

Alcune forme di ROM riprogrammabili contengono dispositivi MOS nei quali è possibile introdurre una quantità di carica permanente (o quasi) nella regione gate-ossido di alcune delle celle, e nulla in altre. Quando la cella deve essere riprogrammata, tutti i dati immagazzinati vengono cancellati esponendo tutta la memoria a una radiazione ultravioletta molto intensa. Dopo la cancellazione, nuovi dati possono venir scritti nella memoria. Questi dispositivi vengono talvolta descritti come **PROM cancellabili** o EPROM **(Erasable PROM)**. Un altro gruppo di ROM indicato come **ROM elettricamente alterabili** (EAROM) possono venire cancellate e ripetutamente riprogrammate per mezzo di metodi elettrici.
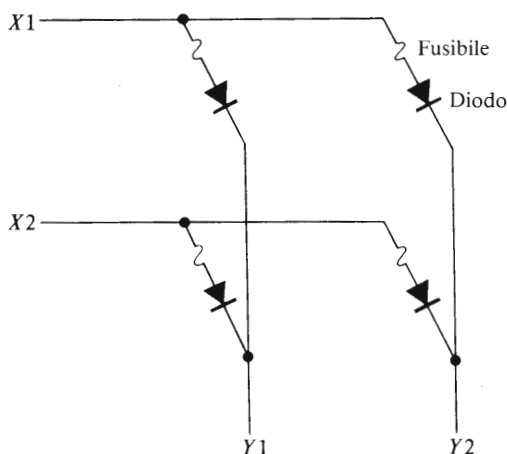
Fig. 9.2 Un tipo di ROM elettricamente programmabile

Le ROM sono ampiamente usate come convertitori di codice (vedi anche il paragrafo 7.7) nei quali l'ingresso alla ROM è rappresentato in un certo codice (per esempio decimale) e l'uscita è in un altro codice (per esempio, binario puro o codice Gray).

Utilizzando le ROM possono venir eseguiti calcoli ripetitivi dato che queste possono immagazzinare tabelle di logaritmi, radici quadrate, fun-

zioni trigonometriche, ecc. Una ROM può essere usata anche per generare una sequenza elettronica e può essere poi utilizzata in applicazioni molto diverse tra loro, in una sequenza di controllo di una macchina utensile o per far funzionare la sequenza di accensione delle luci psichedeliche in una discoteca.

9.4 Dispositivi ad accoppiamento di carica

Il dispositivo ad accoppiamento di carica (CCD charge-coupled device) è un dispositivo MOS a porta multipla che può essere utilizzato per trasferire quantità di cariche tra il terminale di ''source'' e il terminale di ''drain'' sotto il controllo di impulsi multifase.

Il funzionamento dei meccanismi di trasferimento di cariche può essere compreso facendo riferimento alla Fig. 9.3. Il dispositivo in figura ha un substrato di tipo n cosicché l'applicazione di una tensione negativa ad uno qualsiasi dei tre elettrodi metallici del CCD provoca la formazione sotto l'elettrodo di un canale di inversione di materiale di tipo p.
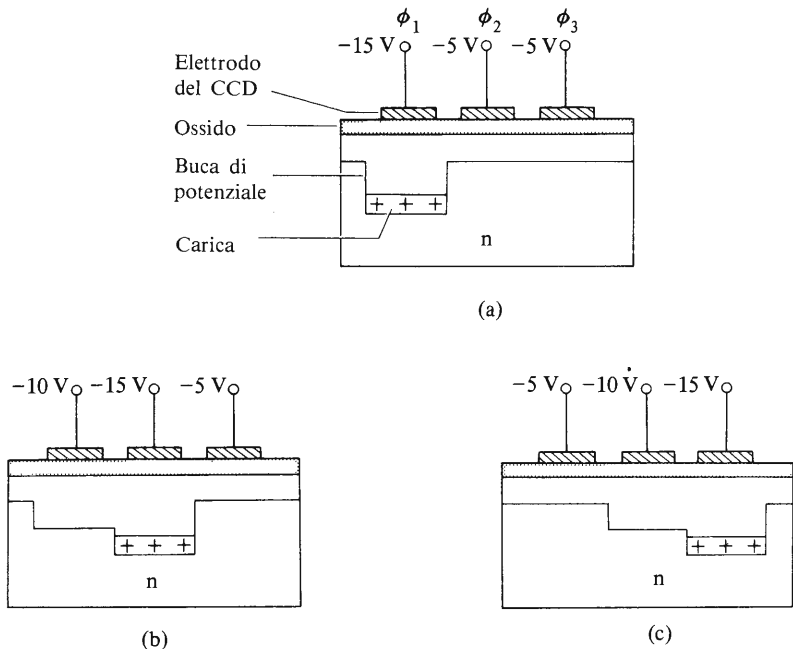
Fig. 9.3 Meccanismi di trasferimento di cariche di un CCD

Un elemento fondamentale CCD ha tre elettrodi ciascuno dei quali viene alimentato da una fase di un clock trifase.

Inizialmente (vedi Fig. 9.3a), all'elettrodo ϕ_1 è applicato il massimo valore di tensione, cosicché la "buca di potenziale" più profonda si forma sotto di esso. Questo fa sì che le cariche positive vengano intrappolate nella buca. Aumentando la tensione dell'elettrodo ϕ_2 e riducendo la tensione dell'elettrodo ϕ_1, il "pacchetto di cariche" si muove fin sotto l'elettrodo ϕ_2 (vedi diagramma b). Applicando la tensione nella sequenza riportata dalla Fig. 9.3, il pacchetto di carica viene trasportato dall'elettrodo ϕ_1 all'elettrodo ϕ_3.

Un registro di scorrimento è formato da una catena in cascata di un certo numero di elementi CCD, in cui i dati vengono inseriti a una estremità della catena di CCD ed estratti dall'altra estremità. Se l'estremità di uscita del registro è collegata all'estremità di entrata, i dati contenuti nel registro vengono fatti circolare in continuazione attorno all'anello che si forma. Questa è una delle modalità con cui possono venire immagazzinate le informazioni in un sistema CCD.

Capitolo 10

Visualizzatori alfanumerici

Una gran varietà di dispositivi visualizzatori alfanumerici viene utilizzata in elettronica e include diodi emettitori di luce (LED), visualizzatori a cristalli liquidi (LCD), tubi a scarica nei gas, tubi a filamento, diodi al fosforo, visualizzatori a tubi a raggi catodici.

10.1 Diodi emettitori di luce

I diodi emettitori di luce (LED = light-emitting diodes) sono dispositivi semiconduttori a giunzione p-n che emettono luce quando sono polarizzati direttamente. I materiali semiconduttori utilizzati sono basati sull'arseniuro di gallio e i colori che vengono generati includono il rosso, il verde, il giallo e l'ambra. Il colore della luce irradiata dipende non soltanto dal semiconduttore ma anche dal tipo di costruzione del diodo. Quando il diodo è polarizzato direttamente, la caduta di potenziale ai suoi capi varia generalmente tra 2 e 2,5 V per una corrente compresa tra 10 e 80 mA. Nel circuito della Fig. 10.1 R è una resistenza che limita il valore della corrente. Il valore di R può essere calcolato mediante la relazione

$$R = \frac{V_S - V_F}{I_F}$$

dove V_S è la tensione continua fornita dal generatore, V_F è la caduta di potenziale ai capi del diodo quando sta conducendo e I_F è la corrente diretta attraverso il diodo.

Se si utilizza un generatore di tensione a 12 V e se la caduta di potenziale ai capi del diodo polarizzato direttamente è di 2 V, il valore di R che limita la corrente a 20 mA è

$$R = \frac{12 - 2}{20 \times 10^{-3}} = 500\ \Omega$$
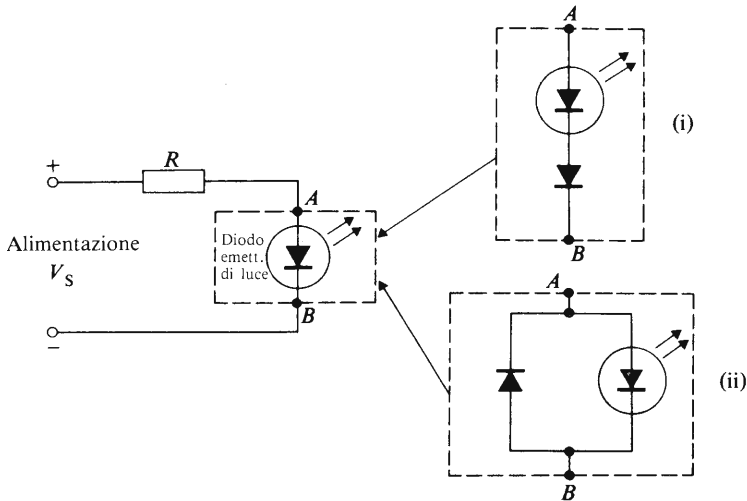
Fig. 10.1 Circuiti a diodi emettitori di luce

In pratica si preferisce utilizzare un valore di 470 Ω ± 10%. La potenza dissipabile dalla resistenza dovrebbe essere maggiore di

$$I_F^2.R = (20 \times 10^{-3})^2 \times 500 = 0{,}2 \text{ W}$$

Una potenza appropriata potrebbe essere di 0,25 W o superiore.

Il valore di tensione di rottura inversa di un diodo emettitore di luce varia generalmente tra i 3 e i 10 V, cosicché quando il diodo emettitore di luce è utilizzato con un generatore di tensione alternata deve essere protetto dalla possibilità di una rottura inversa. I circuiti.impiegati a tale scopo sono riportati nei riquadri (i) e (ii) della Fig. 10.1. Nel riquadro (i) viene incluso in serie al LED un diodo convenzionale. Nel riquadro (ii) un diodo convenzionale è collegato anti parallelamente con un diodo emettitore di luce; quando si applica al diodo emettitore di luce una tensione inversa, il diodo convenzionale conduce e limita la tensione inversa ai capi del diodo emettitore di luce a circa 0,7 V.

I LED possono essere utilizzati individualmente come indicatori luminosi o possono essere opportunamente assemblati per fornire un visualizzatore numerico o alfanumerico.

10.2 Visualizzatori a LED a segmenti

La forma più comune di visualizzatore è il **visualizzatore a sette segmenti** riportato nella Fig. 10.2. Illuminando una combinazione di segmenti come mostrato in Fig. 10.2(b) è possibile illuminare una cifra decimale. Può essere ottenuta anche una limitata varietà di caratteri alfabetici, alcuni dei quali sono riportati nella Fig. 10.2(c).
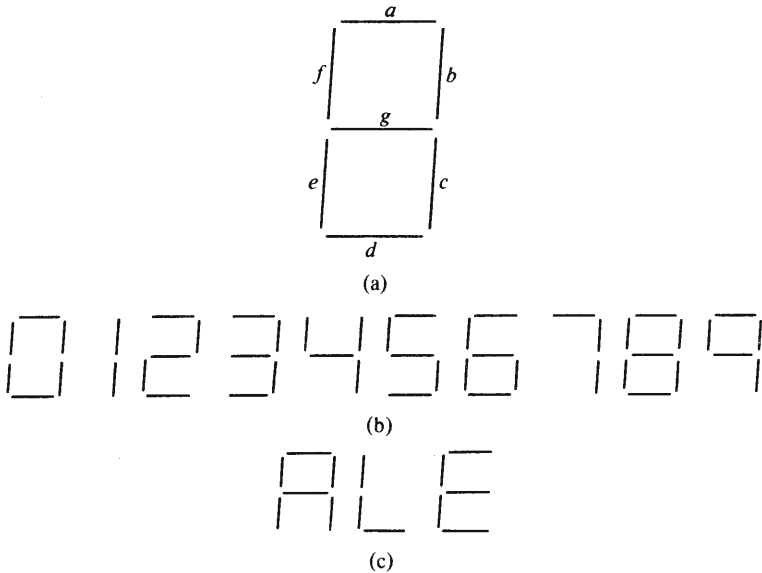
Fig. 10.2 (a) Visualizzatore a sette segmenti, (b) visualizzatore dei valori numerici, (c) alcuni caratteri alfabetici

Nella Fig. 10.3 si vedono due tipi di visualizzatori o LED a sette segmenti. Un tipo della "prima generazione" è riportato nella Fig. 10.3(a); in questo tipo, gruppi di singoli diodi emettitori di luce sono disposti nel substrato in un tracciato a sette segmenti. Nel dispositivo della Fig. 10.3(b) per condurre la luce dal diodo emettitore di luce alla superficie del visualizzatore viene impiegata una cosìdetta "guida di luce", costituita da una cavità a forma di cono che contiene vetro epossidico.

Un diagramma a blocchi del sistema visualizzatore a sette segmenti è riportato nella Fig. 10.4. Gli impulsi d'ingresso vengono contati in un
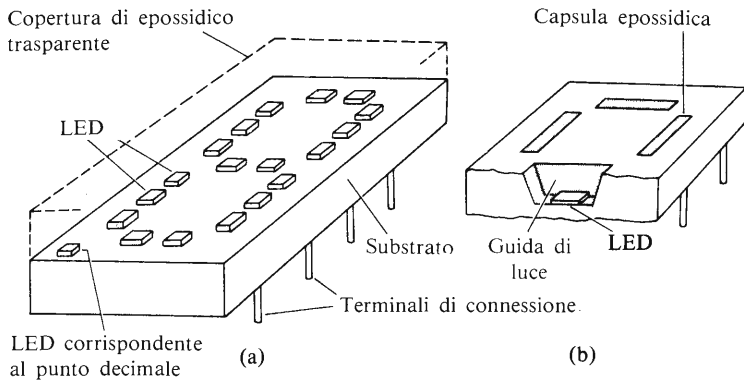
Fig. 10.3 Due tipi di visualizzatori a diodi emettitori di luce a sette segmenti
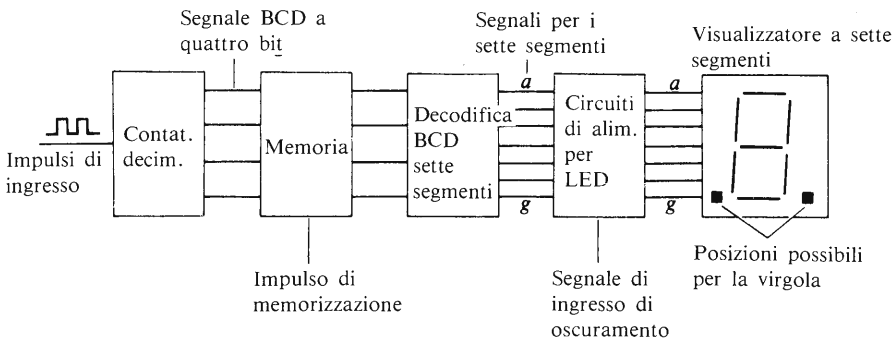
Fig. 10.4 Diagramma a blocchi di un sistema analizzatore a sette segmenti

contatore decimale, e quando il periodo di conteggio è terminato, un impulso viene applicato agli elementi di memorizzazione. Esso provoca il trasferimento del valore immagazzinato dal contatore negli elementi di memoria; questi elementi sono spesso dei flip-flop *D* (vedi paragrafo 5.8) nei quali l'impulso di memorizzazione è l'impulso di clock applicato ai flip-flop. Dopo il completamento dell'impulso di memorizzazione il valore immagazzinato nel contatore compare nella sezione di memoria. Il contatore può allora essere riposizionato a zero e il gruppo successivo di impulsi di ingresso può venire conteggiato. L'aggiunta degli elementi di memoria al sistema permette al valore immagazzinato negli elementi stessi di essere visualizzato in continuazione mentre viene contato il

gruppo meccanico di impulsi; in questo modo il visualizzatore non è "lampeggiante" (o non flickering) eccetto nel momento in cui si verifica l'impulso di memorizzazione quando, cioè, un nuovo valore viene trasferito dal contatore.

Se non è necessaria la proprietà di non lampeggiamento del visualizzatore, si può fare a meno della sezione di memorizzazione.

I segnali sulle linee d'uscita dalla memoria sono applicati alla decodifica che ha sette linee d'uscita, una per ogni segmento (da *a* a *g*) del visualizzatore a sette segmenti. Se è richiesta anche la virgola, il decodificatore deve fornire una ulteriore linea d'uscita per questo scopo. La decodifica può essere progettata come delineato nel paragrafo 7.7 o può essere una memoria a sola lettura.

Per fornire una corrente adeguata ai LED del visualizzatore a sette segmenti, è necessario introdurre un circuito di pilotaggio dei LED tra il decodificatore e il visualizzatore. Molti circuiti di pilotaggio per LED hanno una linea addizionale di ingresso contrassegnata da "ingresso di oscuramento" (blanking input): essa serve perché quando un segnale logico è applicato alla linea di ingresso di oscuramento, provoca l'oscuramento totale dei sette segmenti. Questa proprietà è utilizzata per cancellare tutti gli zeri non significativi nel numero visualizzato. Per esempio se visualizziamo il numero decimale 73 su un visualizzatore a cinque cifre, il visualizzatore indicherebbe 00073; utilizzando la possibilità di oscurare possiamo evitare che i tre zeri non significativi vengano visualizzati.

Esistono altre forme di visualizzatori a segmenti; una forma alternativa è il visualizzatore a sedici segmenti della Fig. 10.5(a). Questo tipo di visualizzatore può fornire la gamma completa dei caratteri alfabetici e numerici, insieme ad alcuni altri caratteri addizionali (vedi Fig. 10.5b).
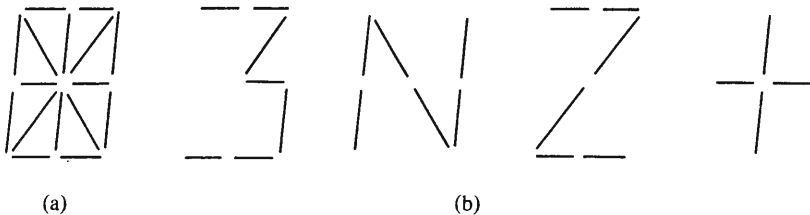
Fig. 10.5 (a) Un visualizzatore a sedici segmenti, (b) alcune delle forme che può assumere

10.3 Visualizzatori a diodi LED a punti (dot array LED displays)

Usando un gruppo di singoli diodi LED in una disposizione a punti di tipo matriciale, è possibile visualizzare l'intera gamma di caratteri alfanumerici insieme con un'ampia gamma di caratteri matematici. Uno dei visualizzatori più comuni di questo tipo è quello a **matrice 5×7** che usa una disposizione di 35 diodi LED disposti in cinque colonne e sette righe.

La struttura del visualizzatore a matrice di punti 5×7 è riportata nella Fig. 10.6. Un diodo LED è collegato tra ciascun filo di riga e ciascun filo di colonna nel punto in cui questi fili si incrociano, come riportato nel riquadro (i) della Fig. 10.6(a). Quando gli interruttori di riga R1 e R2 sono chiusi e anche l'interruttore di colonna C1 è chiuso, i diodi all'intersezione di questi fili sono illuminati. I diagrammi (b) (c) e (d) mostrano alcuni caratteri tipici ottenuti col visualizzatore a punti.

Una disposizione alternativa di punti è quella del visualizzatore 4×7, ma essa possiede una versatilità inferiore a quella del visualizzatore 5×7.
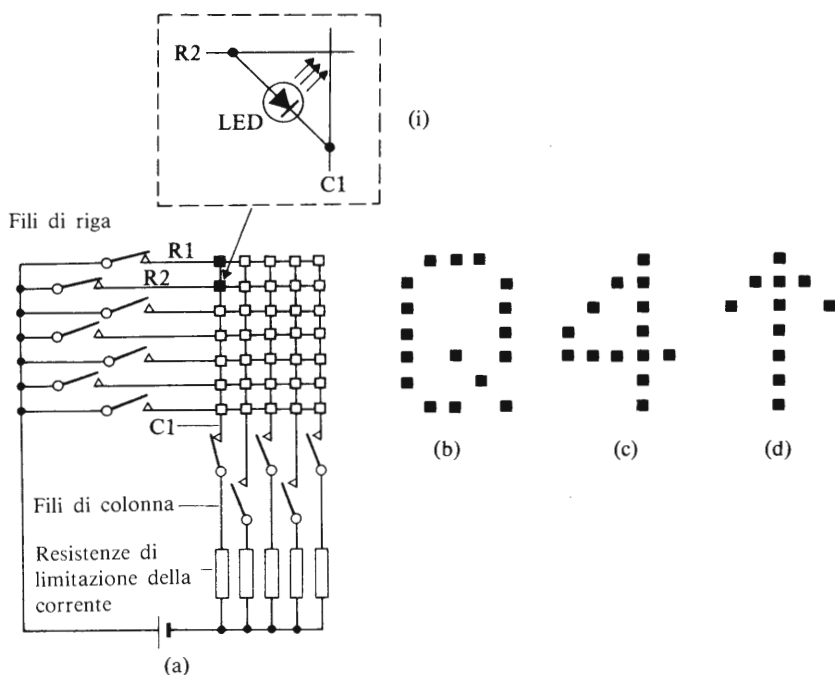
Fig. 10.6 La struttura del sistema di visualizzazione a matrice di punti 5×7

10.4 Visualizzatore a LED multipli

E' utile limitare il numero di circuiti di pilotaggio richiesti per visualizzare un numero elevato, come nel caso di una calcolatrice tascabile, utilizzando la tecnica di multiplessggio (anche nota come **tecnica di scansione**) descritta nel paragrafo 8.6.

La struttura di una forma di multiplessaggio è riportata nella Fig. 10.7. In questo sistema vengono visualizzati due caratteri alfanumerici; il carattere che deve essere visualizzato e la disposizione di diodi LED sulla quale deve venire visualizzato (cioè la disposizione 1 o la disposizione 2) vengono specificati sotto forma di parole binarie sulle linee di ingresso. Questa informazione è immagazzinata nelle memorie temporanee che vengono "indirizzate" dalla parte "indirizzo" della parola binaria d'ingresso. In questo modo, l'informazione da visualizzare nella disposizione è immagazzinata nella memoria temporanea 1 (buffer) ecc. L'informazione in ciascun buffer di immagazzinamento è applicata alla ROM che genera il carattere, in modo che venga visualizzato il carattere corretto.

Per formare un singolo carattere, viene selezionata (o indirizzata) una riga alla volta di diodi LED dal circuito di pilotaggio della riga a cominciare dalla riga superiore. Quando viene selezionata la riga più in alto, i segnali corretti sono forniti dalla ROM attraverso il pilotaggio delle colonne per selezionare i diodi LED della prima riga. Dopo la scansione della prima riga viene indirizzata la seconda riga di diodi attraverso i circuiti di pilotaggio della riga stessa e vengono visualizzati i diodi LED opportuni. Dopo che ciascuna riga è stata indirizzata in sequenza, il ciclo viene ripetuto. Il sistema può essere applicato per trattare un gran numero di diodi emettitore di luce; in questo modo basta una sola ROM per un gran numero di visualizzatori. Un visualizzatore che presenti una buona stabilità di immagine (assenza di "flicker") è ottenuto scandendo ciascun carattere completo almeno cento volte al secondo, cosicché la frequenza di scansione per un visualizzatore di dieci caratteri dovrebbe essere almeno di 1 kHz (1 chilohertz).

10.5 Visualizzatori a cristalli liquidi (Liquid Criystal displays = LCD)

I cristalli liquidi sono materiali liquidi a temperatura ambiente, che hanno una struttura molecolare di tipo filiforme. I cristalli liquidi che vengono utilizzati nei visualizzatori elettronici vengono detti *cristalli liquidi nematici* (dalla parola greca *nematos* che significa filamentoso, filiforme).

L'applicazione di una tensione elettrica attraverso uno stato sottilissi-
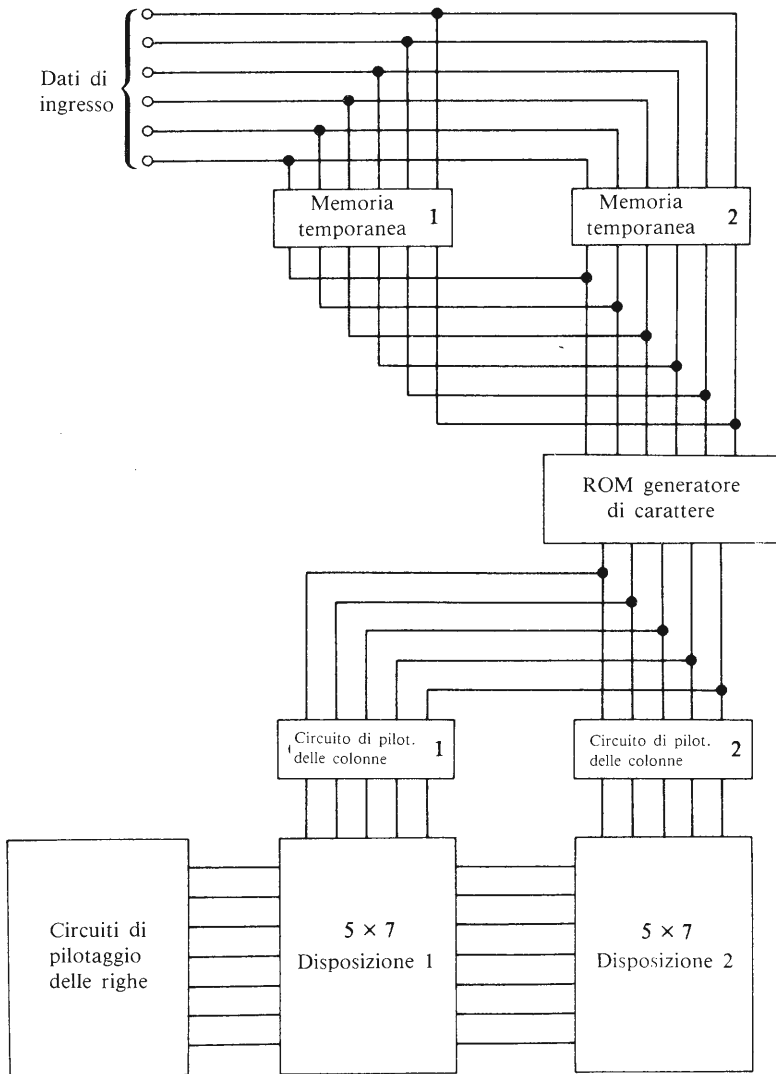
Fig. 10.7 Visualizzatore multiplo a matrice di punti 5×7

mo (generalmente spesso 10 micron) di cristalli liquidi provoca una alterazione delle proprietà ottiche del cristallo. Sono disponibili due tipi di visualizzatori a cristalli liquidi, detti rispettivamente visualizzatori a ef-

fetto di campo e visualizzatori a diffusione dinamica. Il visualizzatore a cristalli liquidi in sé non produce nessuna illuminazione e il suo effetto ottico dipende interamente dall'illuminazione che lo colpisce da una sorgente esterna.

Quando si applica una differenza di potenziale a un **visualizzatore a effetto di campo**, ne deriva generalmente un assorbimento di luce, da parte delle aree del visualizzatore che assorbono energia, e ciò dà come risultato una visualizzazione locale di colore nero. La Fig. 10.8 mostra un visualizzatore di questo tipo. I visualizzatori a cristalli liquidi a effetto di campo progettati in modo appropriato possono essere utilizzati per fornire una visualizzazione argentea.
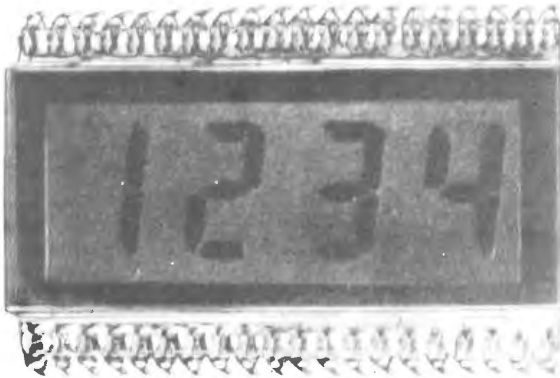
Fig. 10.8 Un visualizzatore a cristalli liquidi

Se si fornisce energia a un **visualizzatore a diffusione dinamica**, certe sue aree diventano "turbolente" diffondendo così efficacemente la luce bianca; questo effetto genera una visualizzazione argentea. In entrambi i tipi di visualizzatori a cristalli liquidi, i segmenti che non vengono eccitati non possono venire distinti dal loro sfondo e sembrano perciò invisibili.

La Fig. 10.9 mostra la struttura fondamentale di un visualizzatore a cristalli liquidi a sette segmenti.

Il cristallo liquido stesso è sigillato tra due superfici di vetro (vedi la Fig. 10.9a), alcune aree del quale hanno una copertura trasparente e conduttrice. Qualsiasi forma di visualizzatore può essere ottenuta in questo modo, ma restringiamo il nostro interesse al visualizzatore a sette segmenti.
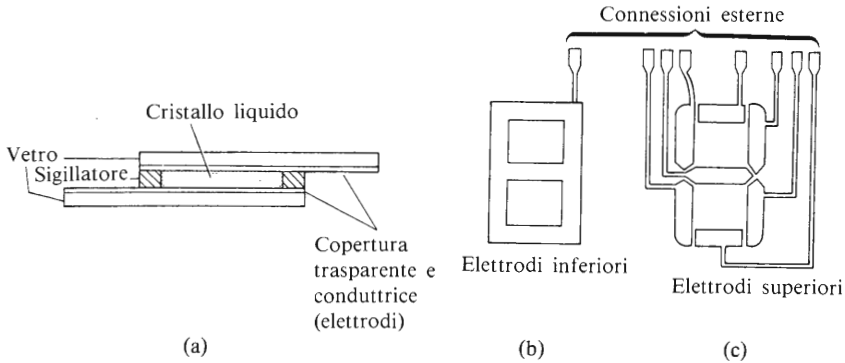
Fig. 10.9 (a) Sezione trasversale di un visualizzatore a cristalli liquidi, (b) elettrodo inferiore, (c) elettrodo superiore

L'elettrodo inferiore (b) è comune a tutti i segmenti, mentre agli elettrodi superiori (c) viene fornita energia indipendentemente gli uni dagli altri (a seconda del numero che deve essere visualizzato); i due insiemi di elettrodi sono allineati uno sopra l'altro nella unità di visualizzazione.

La tensione richiesta per dare energia al visualizzatore varia generalmente tra i 5 e i 16 V efficaci e la potenza consumata è molto bassa (circa 10-15 μW per ogni visualizzazione a sette segmenti).

La ragione di questo basso consumo di potenza è che il visualizzatore a cristalli liquidi non genera nessuna illuminazione per se stesso. Quando si ha bisogno di vedere il visualizzatore a cristalli liquidi al buio è necessario illuminare il visualizzatore per mezzo di una lampada. Il tempo di uso dei visualizzatori a cristalli liquidi è valutato attorno alle 50.000 ore.

Quando viene fornita energia da un generatore di tensione continua, il materiale del cristallo liquido è soggetto ad accorciamento del suo ciclo di vita utile a causa di ciò che è nota come dissociazione elettrolitica del materiale cristallino. Per superare questo inconveniente, viene fornita al visualizzatore a cristalli liquidi energia per mezzo di un generatore di tensione alternata con una frequenza tra i 30 e i 100 Hz generata elettronicamente.

10.6 Visualizzatori a scarica nei gas

Alcuni tipi di visualizzatori a scarica nei gas contengono dieci catodi (ciascun catodo avente la forma di un numero compreso tra zero e nove) e un anodo in un tubo di vetro riempito di gas (generalmente neon).

Quando uno dei catodi è eccitato presenta una luminescenza di un caratteristico colore rosa-arancio.

Questi visualizzatori sono talvolta difficili da leggere in condizioni di elevata illuminazione dell'ambiente, e hanno un angolo di visuale limitato.

Entrambi i visualizzatori a sette segmenti e a matrice di punti sono disponibili sotto forma di visualizzatori a scarica e forniscono caratteri molto luminosi e un ampio angolo di visuale.

I tubi a scarica nei gas richiedono una tensione di alimentazione compresa tra i 160 e i 200 V, ideale per strumenti fissi ma meno pratica per apparati portatili.

Oggi sono disponibili molti circuiti integrati di pilotaggio dei catodi dei tubi a gas da usare in connessione con circuiti logici a semiconduttori; un diagramma a blocchi di uno di essi è riportato nella Fig. 10.10: i segnali da un codice BCD 8421 vengono applicati a circuiti di decodifica
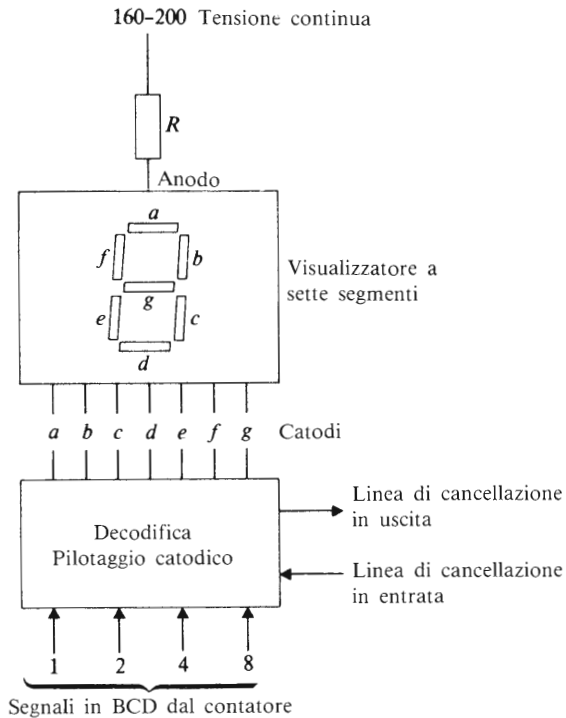
Fig. 10.10 Circuito di pilotaggio per visualizzatore a scarica nei gas

e montaggio (DECODER/DRIVER), dove il codice BCD è convertito in un altro codice adatto per comandare i transistori di pilotaggio dei sette segmenti. Le linee BI e BO sono rispettivamente le linee di ''cancellazione in entrata'' e di ''cancellazione in uscita'', (blanking input, blanking output) che permettono di oscurare il visualizzatore se vi è presente uno zero non significativo (vedi il paragrafo 10.2).

10.7 Visualizzatori a filamenti

Una versione comune del visualizzatore a filamenti usata in accoppiamento con i circuiti logici è il visualizzatore a sette segmenti. Ciascun tubo del visualizzatore ha sette filamenti montati su una piastra rigida nel modo illustrato dalla Fig. 10.2(a), con la piastra inserita in un contenitore di vetro. Come negli altri visualizzatori a sette segmenti il carattere dipende dalla combinazione dei filamenti che vengono illuminati.

La corrente assorbita da ciascun segmento dipende dalla luminosità desiderata e dalle dimensioni del visualizzatore, ed è generalmente compresa tra gli 8 e i 25 mA. La durata di questi dispositivi viene stimata a più di 100.000 ore di vita.

10.8 I visualizzatori a diodi al fosforo

I visualizzatori a diodi al fosforo (fluorescenti) sono basati sul principio dell'''occhio magico'' (indicatore di sintonia) e presentano un colore verde caratteristico quando attivati.

Essi utilizzano piastre anodiche della forma del simbolo da visualizzare, per esempio della forma di ognuno dei sette segmenti dei visualizzatori a sette segmenti. Le piastre anodiche sono rivestite di materiale fosforescente e quando vengono attivate da un generatore sono bombardate con elettroni provenienti da un filamento; l'energia liberata dal bombardamento provoca l'illuminazione dell'elemento. Questi visualizzatori richiedono una tensione anodica di circa 20 V e una tensione di filamento di circa 1,5 V.

10.9 Visualizzatori a tubi a raggi catodici

Forse il tipo di dispositivo visualizzatore più versatile è il tubo a raggi catodici, col quale tutti i lettori hanno familiarità nella sua forma di tubo televisivo. Un'**unità di visualizzazione su video** (VDU = video display

unit) è semplicemente una versione specializzata di un ricevitore televisivo a cui viene connessa una tastiera di macchina per scrivere come dispositivo di interfaccia tra l'operatore e il sistema digitale. I caratteri che possono venir prodotti sulla superficie frontale di un tubo a raggi catodici (CRT = Cathode- ray tube) sono molti e diversi, e restringeremo qui la nostra attenzione al caso del modello a disposizione di punti 5 × 7.

Per familiarizzare il lettore col tubo a raggi catodici, esaminiamo la Fig. 10.11. In essa il catodo K del tubo è riscaldato da un filamento percorso da corrente, che provoca l'emissione di elettroni da parte del catodo. Questi elettroni vengono sottratti dal potenziale positivo del sistema anodico contenente tre anodi A1, A2 e A3. Il sistema anodico è cilindrico cosicché gli elettroni lo attraversano dirigendosi verso lo schermo. Il
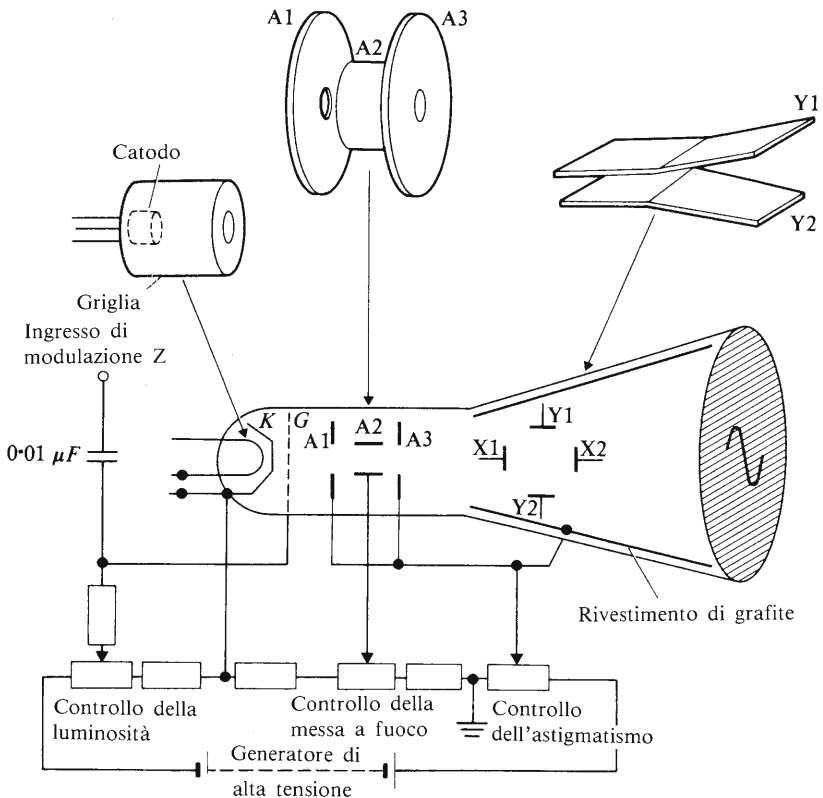
Fig. 10.11 Tubo a raggi catodici

fascio di elettroni è focalizzato sulla superficie frontale del tubo variando la tensione dell'anodo A2 per mezzo del *controllo del fuoco* (focus control). La luminosità del punto sulla superficie del tubo è controllata regolando l'intensità (2) del fascio per mezzo del *controllo di luminosità* (brilliance control).

Prima di raggiungere la superficie frontale del tubo, gli elettroni passano attraverso i piatti di deflessione X e Y (che verranno meglio specificati più avanti), e finalmente colpiscono la superficie del tubo dove cedono la loro energia.

La superficie interna del tubo è rivestita di fosforo che brilla quando gli elettroni lo colpiscono. Il colore del visualizzatore dipende dal tipo di fosforo: i colori tipici sono il bianco, il verde, il rosso e il blu. Gli elettroni ritornano infine al generatore per mezzo del rivestimento di grafite interno al collo del tubo a raggi catodici. La funzione del *controllo di astigmatismo* (astigmatism control) è quella di assicurare che il punto luminoso rimanga circolare in tutti i punti della superficie del tubo (non tutti i tubi a raggi catodici hanno questo tipo di controllo). In una unità di visualizzazione video, i controlli di luminosità, fuoco e astigmatismo sono prefissati e non sono generalmente disponibili per l'utilizzatore.

I segnali di controllo vengono applicati al tubo nella maniera seguente. Il punto luminoso può venir deflesso nella direzione X (cioè orizzontalmente) applicando una tensione tra le placche X1 e X2. Il punto luminoso può essere deflesso nella direzione Y (cioè verticalmente) mediante una tensione tra le placche Y1 e Y2. Inoltre è possibile interrompere il fascio di elettroni con una tensione negativa all'ingresso di modulazione Z.

Utilizzando quest'ultima possibilità, quando il segnale Z di modulazione è zero volt, compare un punto luminoso sulla superficie dello schermo e il punto scompare ogni volta che viene applicata una tensione negativa al terminale Z di modulazione. Applicando contemporaneamente una sequenza di tensioni adeguate agli ingressi X e Y ed una forma appropriata di modulazione Z, si può illuminare una rappresentazione di una matrice di punti sullo schermo del tubo a raggi catodici.

Studiamo ora un metodo per visualizzare un carattere con una matrice di punti 5×7: sia il visualizzatore della lettera A in Fig. 10.12(a); in primo luogo, abbiamo bisogno dei segnali X e Y che provocano il movimento della posizione del raggio in successione a partire dall'angolo in basso a destra del carattere visualizzato attraverso tutte le cinque posizioni "puntuali" lungo la riga inferiore; dobbiamo ora muovere il raggio di una riga verso l'alto, dopo di che esso deve scandire una dopo l'altra tutte le posizioni della seconda riga. Il raggio viene ora di nuovo
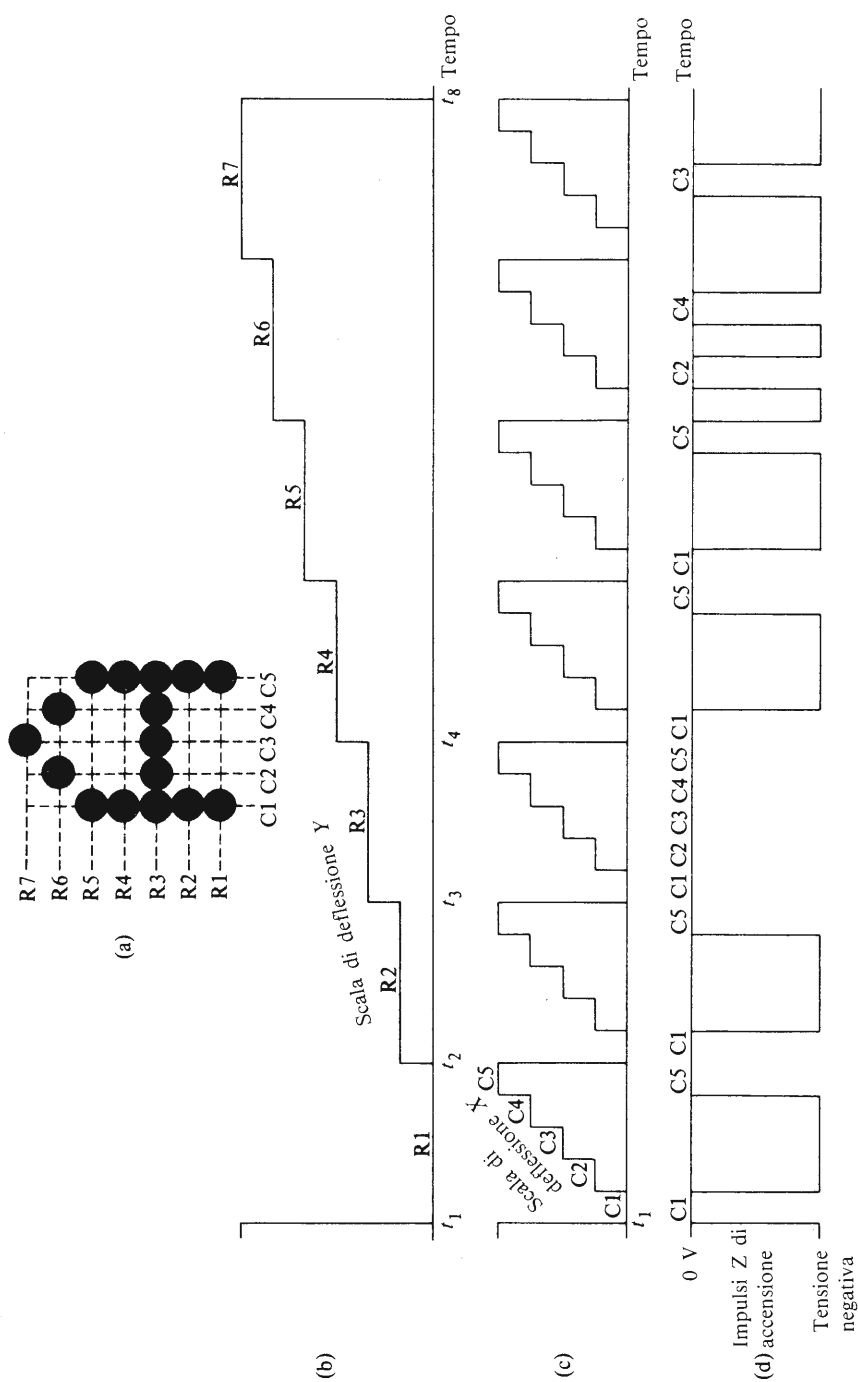
Fig. 10.12 Visualizzatore a matrice di punti 5×7 che utilizza il tubo a raggi catodici

spostato di una riga verso l'alto fino a che tutte le trentacinque posizioni "puntuali" del carattere sono state passate in rassegna. Nello stesso tempo, abbiamo bisogno di applicare un certo tipo di segnale alla modulazione Z, così da illuminare solo quelle posizioni "puntuali" che generano la lettera A.

Si può sfruttare la **forma di onda a scala** (b) e (c) della Fig. 10.12 per provocare la deflessione del raggio attraverso tutte le posizioni della matrice. Al tempo t_1 la tensione di deflessione Y è zero cosicché il raggio è diretto verso la riga più bassa della matrice. Quando la tensione di deflessione X è zero (al temo t_1), il raggio è diretto verso la colonna C1; il primo gradino della forma d'onda di tensione di deflessione X fa sì che il raggio venga diretto verso la colonna C2 e così via, fino a che il raggio non è diretto da ultimo verso la colonna C5.

Al tempo t_2 viene aumentata di un gradino la tensione di deflessione Y, cosicché il raggio è traslato verso l'alto fino alla seconda riga; nello stesso tempo la tensione di deflessione X è riportata a zero, per cui il raggio si trova all'intersezione tra la riga R2 e la colonna C1. Ancora una volta la forma d'onda a scala della X fa sì che il raggio elettronico venga diretto verso una colonna per volta della riga R2.

Il lettore noterà che la tensione di modulazione Z è ridotta a zero quando il raggio elettronico è diretto verso le colonne C1 e C5 e ha un valore negativo (il raggio viene bloccato) quando è diretto verso le colonne C2, C3 e C4. Il risultato è che appare un punto luminoso all'intersezione di R1 con le colonne C1 e C5. Quando il raggio elettronico è diretto verso la seconda riga, i punti che vengono illuminati sono di nuovo quelli che si trovano alle intersezioni di R2 con le colonne C1 e C5. Nella riga R3 (tra il tempo t_3 e il tempo t_4), tutti i punti lungo la riga vengono illuminati.

Un'analisi delle forme d'onda mostra che al completamento della riga R7 (al tempo t_8), il carattere è stato completamente scandito. L'organizzazione e la sequenza temporale di un sistema pratico, potrà naturalmente essere differente da quella descritta in questa sede.

Capitolo 11

Calcolatori e microprocessori

11.1 Cos'è un calcolatore digitale?

Quando parliamo di calcolatore senza nessun'altra specificazione circa il suo uso intendiamo sempre riferirci ad un **calcolatore digitale a programma memorizzato** (stored-program digital computer); (la scrittura americana "program" della parola "programma", in inglese "programm", è universalmente adottata nel mondo dei calcolatori anche di lingua inglese). In termini semplici, possiamo paragonare un calcolatore digitale ad una macchina calcolatrice molto sofisticata che funziona in modo tale che non solo può risolvere problemi complessi ma anche trattare lavoro di routine.

Un diagramma a blocchi di un calcolatore digitale è riportato nella Fig. 11.1, in cui le frecce sul diagramma indicano il flusso di dati e di istruzioni all'interno della macchina. Il cuore del calcolatore è la **unità di elaborazione centrale** (Central processing unit = CPU) che contiene tre sezioni, note rispettivamente come l'unità aritmetico-logica (ALU), l'unità di controllo (Control unit) e la memoria (Store).
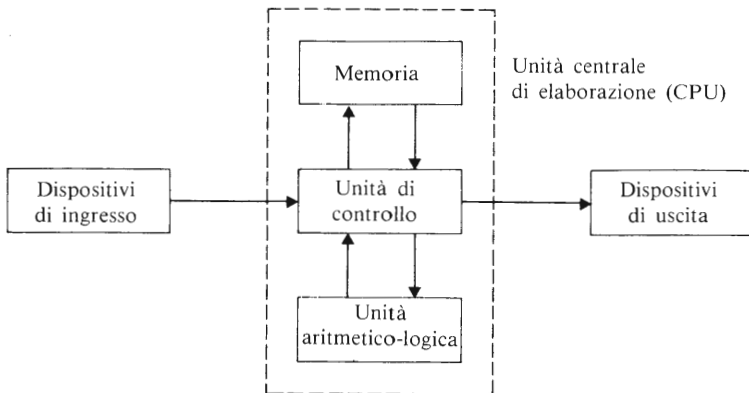
Fig. 11.1 Diagramma a blocchi di un calcolatore digitale

L'unità di controllo è la parte del calcolatore che controlla la sequenza di funzionamento del calcolatore e fa sì che gli eventi si verifichino nell'ordine corretto. Un esempio è riportato più sotto per la sequenza dell'*addizione*.

L'unità aritmetico-logica (ALU) è la parte in cui possono venire eseguite operazioni matematiche (+, —, ×, ÷) e operazioni logiche (AND, OR, NOT) sugli **operandi** (numeri) immagazzinati nel calcolatore. Una parte importante dell'unità aritmetico-logico è nota come l'*accumulatore* ed è lo spazio di lavoro della ALU.

Quando eseguiamo un'operazione sugli operandi, per esempio l'operazione di ADDIZIONE, i dati per mezzo dell'unità di controllo devono essere prelevati nell'unità di memoria e posti nell'accumulatore. L'unità di controllo deve a questo punto prendere il numero da addizionare da un altro indirizzo della memoria e deve far sì che il secondo numero venga addizionato al numero già contenuto nell'accumulatore. Il risultato dell'operazione è alla fine immagazzinato nell'accumulatore, e il suo valore viene a rimpiazzare il numero originariamente posto nell'accumulatore all'inizio dell'operazione. Infine, sotto il controllo del programma del calcolatore, l'unità di controllo deve far sì che il risultato venga copiato dall'accumulatore in un altro indirizzo della memoria.

Nella **memoria** o **magazzino** del calcolatore vengono immagazzinati operandi e istruzioni che devono essere utilizzate nel programma che il calcolatore deve eseguire. Un **programma** è una sequenza di **istruzioni** stilate per la soluzione di un problema particolare. Sotto la direzione dell'unità di controllo, le istruzioni sono eseguite in sequenza, cosicché l'esecuzione del programma è ordinata. La memoria può essere non volatile o volatile (vedi anche il paragrafo 9.1).

Sia i dati che le istruzioni possono perciò essere immagazzinati su nastro magnetico o su disco o su nuclei di ferrite (memoria a nuclei magnetici) o anche, in alternativa, può venire utilizzato uno qualsiasi degli elementi semiconduttori di memoria del tipo descritto nel capitolo 9. Nei calcolatori di medie e grandi dimensioni la maggior parte dei dati è immagazzinata su nastri magnetici o su dischi o in memorie a nuclei magnetici. I sistemi di grandi calcolatori utilizzano una memoria ad alta velocità nell'unità centrale di elaborazione, la quale è sostenuta da una memoria di massa di grande capacità (generalmente su disco magnetico) funzionante ad una velocità inferiore rispetto a quella della memoria centrale veloce.

Nei tipi più piccoli di calcolatori, noti come microcalcolatori (vedi anche il paragrafo 11.4), la maggior parte delle informazioni è immagazzinata su memorie a semiconduttori. Molte memorie a semiconduttori per-

dono i dati immagazzinati quando la tensione cade a zero (memorie volatili); a meno che non vengano prese particolari precauzioni contro l'eventualità di una interruzione nella fornitura di tensione, in questa circostanza la memoria perderebbe non solo i dati su cui sta lavorando ma anche il programma stesso.

I dati e le istruzioni possono essere inseriti nel calcolatore per mezzo di una vasta gamma di **dispositivi di ingresso** che comprendono interruttori e sensori, lettori di nastri di carta perforati, telescriventi (una telescrivente è una specie di macchina per scrivere elettrica che genera un modello di codice binario accettabile da tutti i calcolatori), registratori di nastri, schede magnetiche ecc.

Il calcolatore comunica con il mondo circostante per mezzo di **dispositivi di uscita** che comprendono lampadine, diodi emettitori di luce, telescriventi, plotters, perforatrici di nastri, unità di visualizzazione video, ecc. La maggior parte delle installazioni di calcolatori di grandi e medie dimensioni hanno anche un dispositivo di uscite noto come **stampante per linee**; questa può essere considerata come una macchina per scrivere elettrica ad altissima velocità che stampa una linea completa del testo nello stesso tempo in cui una macchina per scrivere stampa un singolo carattere. La stampante per linee, stampa i risultati ad una velocità molto più elevata di quella di una telescrivente, ma il costo iniziale e il costo di funzionamento sono più elevati.

Quando all'utilizzatore serve avere una copia materiale (cioè *una copia stampata*; in inglese hard copy) del risultato, deve essere impiegata una telescrivente o una stampante per linee (o un altro dispositivo di uscita in qualche modo equivalente).

Quando invece non è necessaria una copia stampata, una **unità di visualizzazione su video** (VDU = Video display unit) può essere più economica, dato il costo inferiore di visualizzazione dei risultati.

I dispositivi di ingresso e uscita sono sinteticamente indicati come **dispositivi periferici o periferiche**.

Per completare questo paragrafo introduttivo, pensiamo che il lettore troverà utile un accenno al significato delle parole "hardware" e "software". L'**hardware** di un calcolatore consiste nei componenti elettronici utilizzati nella costruzione del calcolatore, cioè gli oggetti che possono venir toccati materialmente.

Il **software** consiste nei programmi del calcolatore, scritti dall'utilizzatore, che permettono al calcolatore di eseguire le istruzioni. In una zona intermedia tra le definizioni di hardware e software risiede un certo quantitativo di istruzioni note come **firmware**. Queste istruzioni sono collocate in una parte della memoria su una memoria di sola lettura

(ROM) e vengono programmate durante la costruzione o sul posto di utilizzazione da un tecnico.

L'utilizzatore (o il programmatore) non ha accesso al firmware.

11.2 Diagrammi di flusso

Quando siamo di fronte ad un problema, abbiamo bisogno di coordinare i nostri pensieri per decidere come risolverlo. Un metodo è quello di rappresentare il processo di risoluzione del problema sotto forma di diagramma; quando esso viene presentato in questa forma, è noto come diagramma di flusso. Alcuni dei simboli utilizzati nei diagrammi di flusso sono riportati nella Fig. 11.2.
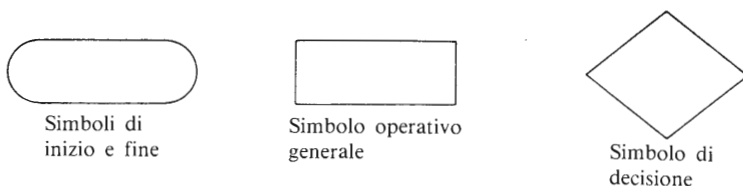
Fig. 11.2 Alcuni simboli presenti nei diagrammi di flusso

Per illustrare i principi fondamentali della stesura di un diagramma consideriamo un esempio di programma per il calcolo della corrente in un circuito contenente N resistenze in serie, dove il valore di N è un qualsiasi numero intero specificato dall'utilizzatore del programma. L'utilizzatore ha bisogno di introdurre nella memoria del calcolatore i seguenti dati:

1. la tensione di alimentazione
2. il numero N delle resistenze nel circuito
3. il valore di ciascuna resistenza.

Infine il calcolatore deve calcolare il valore di corrente e visualizzare il risultato su un dispositivo di uscita. Un possibile metodo di soluzione è riportato nella Fig. 11.3.

Nella soluzione proposta assegnamo una cella della memoria (designata dal simbolo R) per immagazzinare la somma delle resistenze dei vari resistori. Abbiamo bisogno di specificare la tensione fornita, V, e il numero delle resistenze N nel circuito, dove il valore di N viene immagazzinato in una cella particolare denominata X. Il valore N è assegnato a un
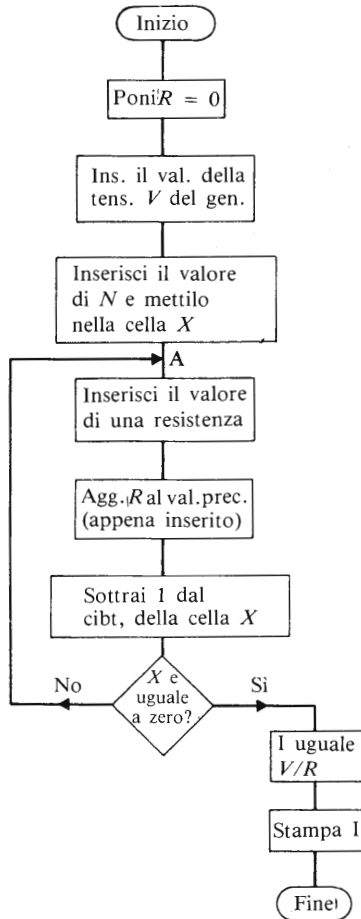
Fig. 11.3 Semplice diagramma di flusso

contatore "all'indietro" (vedi anche il paragrafo 7.3). Assumiamo che la tensione del generatore sia 20 V e che ci siano tre resistenze nel circuito ($N=3$).

In un punto A del diagramma di flusso entriamo in ciò che è noto come un **ciclo** (loop) del programma. Il primo passo all'interno è quello di domandare all'operatore di inserire dalla tastiera della telescrivente (o dal tipo di dispositivo di ingresso utilizzato) il valore della prima resistenza. Assumiamo che sia di 10 Ω (10 ohm). Il valore immagazzinato

nella cella *R* è quindi aggiunto a questo valore, cosicché il valore immagazzinato nella cella *R* diventa 10 Ω.

Avendo fatto questo sottraiamo 1 dal valore della cella *X* (che era originariamente 3), lasciando il valore 2 nella cella *X*. Nell'elemento decisionale ci domandiamo se il contenuto della cella *X* è zero; poiché la risposta in questo caso non è negativa, siamo diretti ancora una volta all'inizio del ciclo in A.

L'operatore inserisce ora il valore della seconda resistenza; se questo è di 6 Ω, allora il prossimo passo fa sì che questo valore venga addizionato al contenuto della cella *R*; il valore immagazzinato in R è ora $10+6=16$ Ω.

Il contenuto della cella *X* è ridotto ancora una volta di 1 (e fornisce $X=1$), e l'elemento decisionale ci dirige di nuovo all'inizio del ciclo. L'operatore inserisce allora il valore della terza resistenza; se questo valore è 4 Ω, il passo successivo del programma fa sì che il contenuto della cella *R* diventi $16+4=20$ Ω. Quando sottraiamo 1 dal contenuto di *X* raggiungiamo la condizione $X=0$. Il programma può quindi uscire dal lato destro del simbolo decisionale, dove il calcolatore trova il valore della corrente del circuito ($I=10$ V/20 $\Omega=1$ A) e visualizza il risultato su un dispositivo di uscita.

11.3 I linguaggi di programmazione

Nei primi stadi di sviluppo dei calcolatori digitali era necessario istruire il calcolatore nel solo linguaggio che esso era in grado di capire — cioè col codice binario. Questo linguaggio permette al programmatore di inserire le istruzioni e i dati direttamente nella memoria del calcolatore. Quando è scritto in questa forma, il programma viene definito in **linguaggio macchina**. Un linguaggio macchina è detto anche **linguaggio di basso livello**. Qualsiasi tipo di linguaggio per il calcolatore venga utilizzato dal programmatore, le istruzioni devono in ultima analisi essere convertite nel linguaggio macchina prima che il calcolatore le possa comprendere.

Un altro tipo di linguaggio di basso livello è noto come **linguaggio assemblatore** (Assembly language). Ciascuna istruzione nel linguaggio assemblatore consiste di tre o quattro lettere mnemoniche (cioè "che aiutano la memoria"); alcune di esse per esempio sono LDA, MLT, STA insieme ad una informazione circa un indirizzo. Per esempio LDA significa "carica il contenuto dell'indirizzo specificato nell'accumulatore" ("Load" in inglese = caricare), MLT significa "moltiplica il contenuto

dell'accumulatore per il contenuto dell'indirizzo specificato'' (''Multiply'' in inglese = moltiplicare), e STA significa ''immagazzina il contenuto dell'accumulatore nell'indirizzo specificato (''store'' in inglese = immagazzinare). Supponiamo di trovare la seguente successione di istruzioni in un programma.

LDA 101
MLT 102
STA 103

Queste istruzioni vengono eseguite in sequenza. Il calcolatore digitale carica il contenuto dell'indirizzo 101 nell'accumulatore (questo simultaneamente cancella i dati immagazzinati in precedenza nell'accumulatore), dopo di che il contenuto dell'accumulatore viene moltiplicato per il contenuto della cella di memoria 102, ed il prodotto risultante viene quindi immagazzinato nell'accumulatore. Infine, il valore dell'accumulatore viene trasferito nell'indirizzo 103.

Dato che ciascuna singola istruzione si riferisce solo ad un indirizzo, ciascuna istruzione è una **istruzione ad un solo indirizzo.** L'istruzione ad un solo indirizzo è molto economica in termini di capacità di memoria necessaria per eseguirla. Vengono utilizzate nei calcolatori anche istruzioni a indirizzo multiplo; un esempio di **istruzione a tre indirizzi** è riportata qui sotto.

MLT 101, 102, 103

Essa si legge come segue: moltiplica il contenuto dell'indirizzo 101 per il contenuto dell'indirizzo 102 e immagazzina il risultato nell'indirizzo 103. Questa è chiaramente una istruzione potente, ma è costosa in termini di struttura circuitale del calcolatore necessaria ad immagazzinarla e ad eseguirla. Nella maggior parte delle macchine, le operazioni dell'unità centrale di elaborazione vengono eseguite utilizzando un formato ad un solo indirizzo, ma la possibilità di utilizzare istruzioni ad indirizzo multiplo viene fornita al programmatore per mezo del software o del firmware del calcolatore. La maggior parte dei microcalcolatori e dei microprocessori (vedi il paragrafo 11.4) viene programmata in un linguaggio di basso livello come il linguaggio assemblatore, che viene tradotto nel linguaggio macchina appropriato per mezzo di un programma speciale noto come **assemblatore** (assembler).

Molti **linguaggi d'alto livello** o **linguaggi compilatori** (high level languages, o compiler languages) sono stati elaborati in seguito; questi lin-

guaggi sono il più possibile correlati con la lingua di ogni giorno. Nel caso di un calcolatore che svolge mansioni del tutto generali, questo ha ovviamente i suoi vantaggi; lo scotto che si paga è l'accresciuta complessità, l'accresciuto costo, e una riduzione di efficienza del sistema che ne risulta, quando viene confrontato con un sistema che usa un linguaggio di basso livello. I due linguaggi scientifici più importanti sono il FORTRAN (FORmula TRANslation = traduzione di formule) e l'ALGOL (ALGOrithmic Language = linguaggio algoritmico) (un *algoritmo* è una serie di affermazioni che descrivono il procedimento per la risoluzione dei problemi: il nome deriva dal matematico arabo Alkarismi).

Un linguaggio comune nel mondo commerciale è il COBOL (Common Business Oriented Language = linguaggio orientato verso le attività commerciali). Ancora un altro linguaggio è noto come PL/1 (Programming Language 1 = linguaggio di programmazione 1).

Sono stati sviluppati numerosi linguaggi interattivi mediante i quali il calcolatore può tenere una "conversazione" con l'utilizzatore per mezzo di un dispositivo di entrata e uscita durante il tempo in cui il programma viene preparato.

Per far questo, il programmatore deve rimanere **in linea** con il calcolatore per tutto il tempo in cui egli prepara il suo programma, cioè, il programmatore prepara il suo programma mentre è seduto alla telescrivente o di fronte all'unità di visualizzazione video.

Una caratteristica dei sistemi interattivi è che il calcolatore può avvisare il programmatore di alcuni errori che egli può commettere durante la preparazione del programma.

Forse il linguaggio interattivo più comune è il linguaggio di programmazione BASIC. Illustriamo ora l'uso del linguaggio BASIC per mezzo di un programma che calcola il valore della corrente in un circuito elettrico contenente N resistenze in serie (vedi anche il diagramma di flusso della Fig. 11.3). Eccolo di seguito:

```
LET R=0
INPUT V
INPUT N
FOR C=1 TO N
INPUT S
LET R=R+S
NEXT C
LET I=V/R
PRINT I
END
```

Il programma richiede prima di tutto che una cella di memoria R immagazzini il valore zero. La ragione di questa istruzione è che il programma avrebbe potuto essere stato utilizzato da qualche altro operatore che (come vedremo) lascia i dati in questa cella; questo valore è inutile ai fini del nostro calcolo, e perciò dobbiamo assicurarci che R sia 0 all'inizio del nostro programma.

In seguito si richiede all'utilizzatore di inserire il valore della tensione fornita dal generatore per mezzo del suo dispositivo di ingresso, dopo di che deve inserire il numero (N) di resistenze presenti nel circuito. Il programma quindi entra in un ciclo chiamato FOR-NEXT che comincia con l'affermazione FOR $C=1$ TO N (per C che va da 1 a N incrementando il suo valore di una unità alla volta) e che finisce con NEXT C (ed ora ripeti C fino che sei arrivato a N). Il programma circola attorno a questo ciclo N volte e ciascuna volta aggiunge il nuovo valore della resistenza (S) al valore precedente immagazzinato nella cella R. Supponiamo che N sia 3 e che i valori delle resistenze siano 10 Ω, 6 Ω, e 4 Ω rispettivamente; la successione dei valori immagazzinati nella cella R è 0,10,16,20; il valore finale è il valore della resistenza totale del circuito.

L'istruzione LET $I=V/R$ (poni I uguale a V/R) fa sì che la macchina calcoli la corrente nel circuito, e l'istruzione PRINT I (print = stampa) fa stampare il valore della corrente.

Prima che il calcolatore possa eseguire le istruzioni del linguaggio ad alto livello, ciascuna istruzione deve essere tradotta nel linguaggio-macchina (o assoluto). La traduzione (o transcodifica) è eseguita da un programma detto **compilatore** (compiler), una versione sofisticata del-programma assemblatore menzionato in precedenza. Ciascun sistema calcolatore possiede una gamma di compilatori elaborati su misura per adattarsi ad un linguaggio ad alto livello.

11.4 Microprocessori e microcalcolatori

Un microprocessore (microprocessor; μP) è un singolo circuito integrato che contiene la maggior parte della potenza di calcolo necessaria per un piccolo calcolatore, comprese tutte le funzioni aritmetiche e logiche. Una tipica piastra di un circuito con microprocessore è riportata nella Fig. 11.4, dove il microprocessore è il circuito integrato bianco al centro dello stampato.

Sono necessari altri circuiti integrati per convertire il microprocessore in un **microcalcolatore** utilizzabile (microcomputer). La Fig. 11.5 riporta un diagramma a blocchi di un tipico sistema microcalcolatore. I sistemi
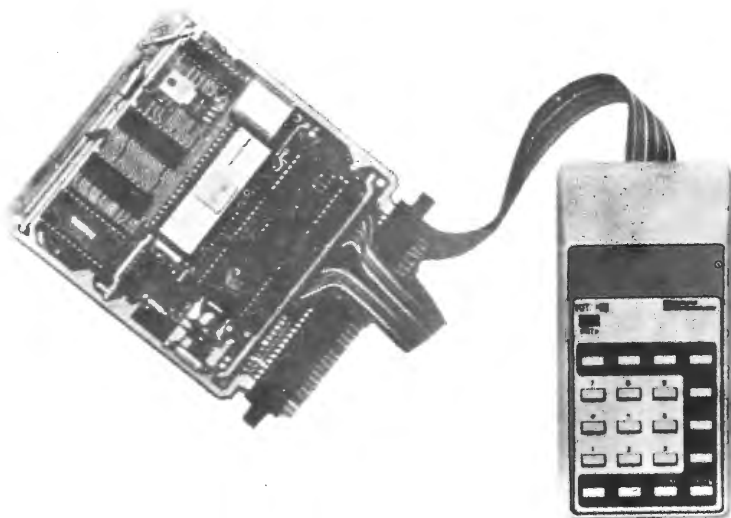
Fig. 11.4 Un piccolo sistema a microprocessore (riprodotto per gentile concessione della National Semiconductor Ltd)

microcalcolatori hanno un'unità centrale a microprocessore montata su un circuito stampato talvolta insieme con la sua alimentazione e col suo oscillatore di sincronismo clock; l'oscillatore può essere controllato con un quarzo consentendo una maggior precisione. Gli impulsi di clock sono utilizzabili non solo per controllare la sequenza di eventi nel microprocessore, ma anche per fornire impulsi con una accurata temporizzazione. Per esempio, un ritardo di tempo noto può essere generato semplicemente contando gli impulsi di clock: un esempio di questo tipo di applicazioni è illustrato nella Fig. 11.8.

Il microcalcolatore è organizzato attorno ad un sistema di **tre insiemi di connessione** detti "**bus**" (pronunc. bas) che permette di collegare tutte le parti del calcolatore. Il **bus di controllo** (control bus) contiene molte linee ed assolve all'importante funzione di informare il sistema del momento in cui leggere o scrivere, o fermarsi o riposizionare (read, write, halt, reset rispettivamente), ecc.

Il bus di indirizzamento (address bus) contiene una gran quantità di linee, generalmente sedici; questo bus permette alla micro unità di elaborazione (MPU = microprocessing unit) di indirizzare la memoria.

Il **bus dei dati** (data bus) contiene abbastanza linee (di solito otto) da permettere ad una parola binaria di essere trasmessa lungo di esso in modo parallelo.
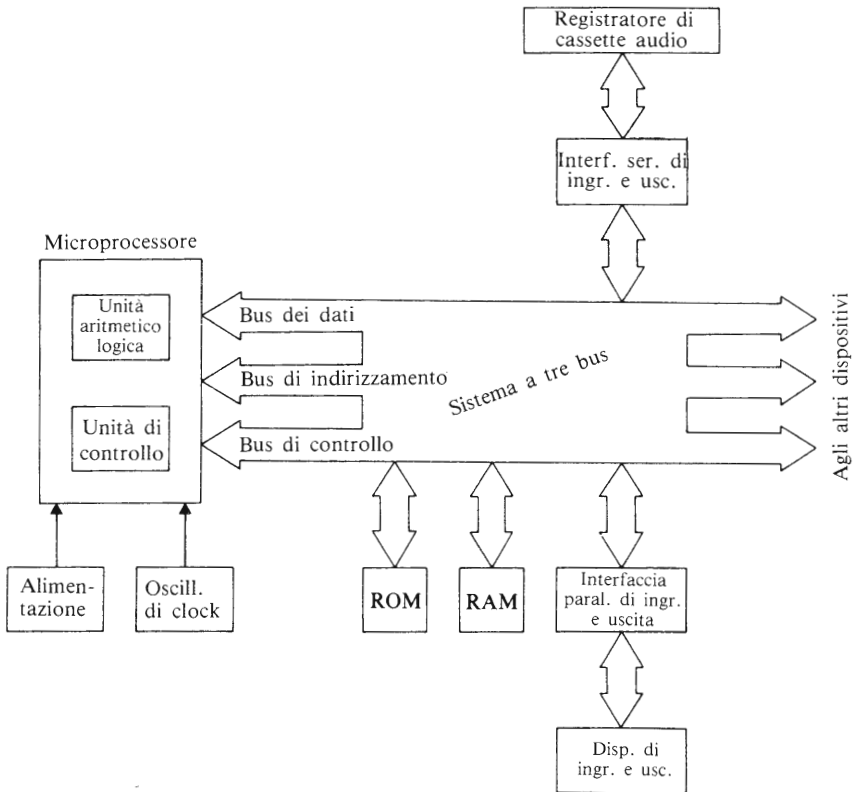
Fig. 11.5 Un sistema microprocessore

La memoria di sola lettura ROM contiene i dati che sono essenziali per il funzionamento del microprocessore; questi dati non devono andare perduti quando si spegne l'alimentazione, sono quindi programmati all'interno della memoria di sola lettura al momento della sua costruzione (vedi anche il paragrafo 9.3).

La memoria ad accesso casuale RAM contiene il programma e i dati su cui dovrà operare il microprocessore.

I sistemi a microprocessore possono essere programmati direttamente con il codice binario puro o più frequentemente con il codice esadecimale (vedi anche capitolo 6). La programmazione in un linguaggio di basso livello di questo tipo è soddisfacente nel caso in cui il programma sia abbastanza semplice. Ciascun microprocessore possiede ciò che è noto co-

me un **insieme di istruzioni** (instruction set) che elenca i passi di programma a disposizione dell'utilizzatore.

Un punto di partenza utile nella selezione di un microprocessore è il confronto tra gli insiemi di istruzioni, dato che un insieme di istruzioni ampio può significare che il programma richiede un numero minore di istruzioni rispetto al caso in cui l'insieme di istruzioni sia limitato. Dato che ogni tipo di micro unità di elaborazione (MPU) ha un suo proprio insieme di istruzioni, non tenteremo qui di descrivere un singolo insieme. Quando un utilizzatore intende sviluppare una particolare applicazione, gli si consiglia di domandare una dimostrazione del sistema di sviluppo di un microprocessore, cosicchè egli possa controllare il software interno.

Quando devono essere realizzati programmi più lunghi, può essere consigliabile l'utilizzo di un sistema a microprocessore che possa trattare sia un linguaggio assemblatore, sia un linguaggio di alto livello. Come abbiamo già visto, ciò richiede ulteriori programmi immagazzinati nella parte di memoria della macchina.

I dispositivi di ingresso e uscita (I/O) sono necessari per comunicare con la micro unità di elaborazione; vengono inoltre richieste opportune unità di interfaccia tra il sistema di tre bus e i dispositivi di ingresso e uscita in modo da permettere un loro corretto accoppiamento.

In generale ci sono due tipi di unità interfaccia, precisamente quelle che accettano ingressi seriali (cioè, un impulso alla volta) e quelle che accettano ingressi in parallelo (cioè, molti impulsi simultaneamente). I terminali di ingresso sono generalmente chiamati **porti di ingresso** (input ports) e i terminali di uscita sono conosciuti invece come **porti di uscita** (output ports).

Un **segnale di ingresso seriale** è applicato lungo una singola linea; pertanto la parola binaria viene presentata al sistema come una sequenza di impulsi. Dato che il microprocessore lavora in modo parallelo, la funzione dell'unità di interfaccia è quella di convertire un segnale seriale in-forma parallela nella quale tutti i bit vengono presentati alla micro unità di elaborazione in parallelo (cioè, simultaneamente) sul bus dei dati. La Fig. 11.5 illustra l'uso di una di queste unità di interfaccia in connessione con un registratore a cassetta. L'unità di interfaccia verrà descritta come un **Modem** (Modulator-DEMmodulator = modulatore-demodulatore) che converte i segnali logici in suoni, cosicché il programma e i dati possono essere immagazzinati in una cassetta a nastro per mezzo di un registratore a basso prezzo. I modem progettati per essere utilizzati con molti sistemi digitali impiegano il sistema CUTS (Computer User Tape Standard), nel quale un segnale digitale "0" è convertito in un

suono di 1200 Hz e un "1" logico in un suono di 2400 Hz.

Quando numerose linee di ingresso presentano segnali simultanei, è necessaria un'interfaccia parallela I/O. Questo può succedere quando un sistema ha sensori in molti punti differenti, o quando i segnali devono alimentare simultaneamente lampade, relè, ecc.

I microprocessori possono essere utilizzati in applicazioni che vanno dal controllo delle macchine utensili al funzionamento dei giochi televisivi, al controllo di accessi ecc. Un vantaggio dichiarato nell'utilizzazione dei microprocessori al posto dei sistemi a logica cablata è che tutto ciò che si deve fare per cambiare l'applicazione del sistema a microprocessore è semplicemente un cambiamento di programma; i componenti rimangono gli stessi.

Un altro vantaggio di un sistema a microprocessori è la sua costruzione modulare, cosicché l'utilizzatore può cominciare con un sistema fondamentale comprendente il chip del microprocessore, le memorie ROM e RAM e un numero limitato di dispositivi di ingresso e uscita. I dispositivi di ingresso possono essere semplicemente sotto forma di interruttori e i dispositivi in uscita possono essere diodi emettitori di luce. Il sistema può essere espanso quanto si vuole, per esempio il dispositivo di ingresso e uscita può essere una telescrivente portatile (vedi Fig. 11.4), una telescrivente completa o una unità di visualizzazione. Naturalmente, anche le dimensioni del sistema a microprocessore devono espandersi in connessione con quelle delle installazioni.

Quando si utilizza un microprocessore per realizzare una calcolatrice programmabile con una unità di visualizzazione video, si richiede un'altra memoria ROM per il generatore di caratteri del video; si richiede inoltre una memoria RAM aggiuntiva per trattare i programmi che devono essere inseriti dall'utilizzatore. Se deve venire utilizzato un linguaggio di alto livello, si richiede una considerevole quantità di spazio di memoria per immagazzinare il programma compilatore. Inoltre un linguaggio d'alto livello ha l'effetto di rallentare la velocità complessiva di funzionamento del microprocessore, dato che ciascuna istruzione ad alto livello richiede un tempo abbastanza lungo per essere eseguita. Di conseguenza i sistemi microprocessori che possono funzionare efficientemente con un linguaggio di alto livello sono costosi da acquistare e da far funzionare quasi quanto un piccolo minicalcolatore. Le applicazioni in cui i microprocessori offrono un considerevole risparmio economico combinato con un'elevata velocità di funzionamento sono quelle in cui essi sono una parte integrante di sistemi automatici. Questi comprendono sistemi nei quali i dati vengono inseriti nelle micro unità di elaborazione direttamente dall'elettronica del sistema anziché dall'operatore, e quelli in

cui l'uscita del microprocessore controlla il sistema.

Un esempio di questo tipo è illustrato nel paragrafo 11.5. I microprocessori offrono inoltre vantaggi quando l'interfaccia tra essi e il mondo umano è relativamente semplice.

11.5 Controllo della sequenza di movimento di due pistoni per mezzo di un microprocessore

Consideriamo ora come realizzare un controllo della sequenza di inserimento di due pistoni nella quale i due pistoni o stantuffi A e B seguono la sequenza A_+, B_+, A_-, B_-. La disposizione dei pistoni è riportata nella Fig. 11.6; la pressione del fluido applicata ai cilindri A e B è controllata da un'elettrovalvola di controllo a due posizioni e quattro vie. Le valvole di controllo sono azionate da segnali preventivi dai porti di uscita della micro unità di elaborazione. I segnali dagli interruttori di limite A_1, B_2, B_1 e B_2 vengono applicati ai porti di ingresso del microprocessore. I pistoni devono essere all'inizio completamente ritratti e la sequenza deve essere iniziata per mezzo di un pulsante di START (marcia); dopo il completamento della sequenza A_+, B_+, A_-, B_-, i pistoni devono restare stazionari finché il pulsante di marcia venga nuovamente premuto.

Illustriamo lo sviluppo del programma per mezzo di un diagramma di flusso, che presenta il modo in cui il microcalcolatore investiga lo stato dei sensori A_1, A_2, B_1 e B_2; gli stati di questi sensori sono applicati in parallelo alla microunità di elaborazione sotto forma di parola binaria. Le parole binarie nei vari punti importanti della sequenza sono elencate
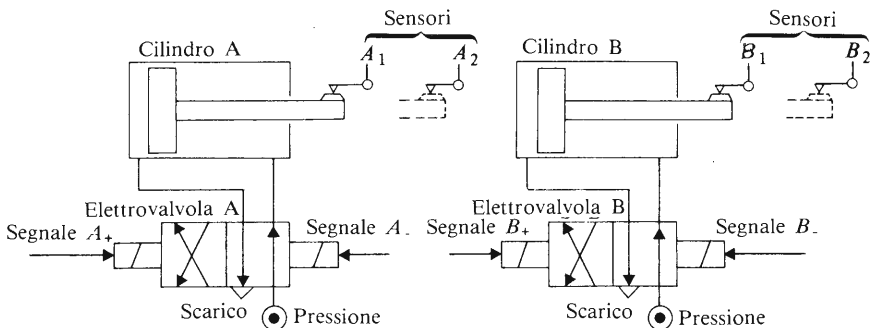
Fig. 11.6 Controllo della sequenza dei pistoni

Tabella 11.1

Stato dei sensori (condizioni di riferimento)				*Posizione dei pistoni*	*Azione da intraprendere*
A_1	A_2	B_1	B_2		
1	0	1	0	A e B ritratti	Uscita $A+$ (a condizione che il pulsante di marcia venga premuto)
0	1	1	0	A estratto, B ritratto	Uscita $B+$
0	1	0	1	A e B estratti	Uscita A_-
1	0	0	1	A ritratto, B estratto	Uscita B_-

nella Tabella 11.1. Indicheremo le parole binarie della colonna di sinistra della Tabella 11.1 come *condizioni di riferimento*.

Guardiamo ora il diagramma di flusso della Fig. 11.7.

Il programma comincia con la lettura degli stati dei sensori $A_1A_2B_1B_2$, quindi sottrae dalla parola binaria letta il valore di riferimento di 1010 (prima riga della Tabella 11.1). Se il risultato di questa operazione è zero, $A_1=1$, $A_2=0$, $B_1=1$, $B_2=0$, cioè i pistoni A e B sono completamente ritratti; in questo caso abbandoniamo l'uscita "SI" del simbolo decisionale D1 e ci muoviamo verso il simbolo decisionale D2. Se il pulsante di marcia non è stato premuto passiamo dall'uscita "NO" del simbolo decisionale D2 e ritorniamo all'inizio del ciclo. Il programma continua ad eseguire questo ciclo fino a che il pulsante di marcia non è stato premuto, e quindi possiamo uscire dall'uscita "SI" del simbolo decisionale D2. Questo fa sì che venga applicato un segnale di un "1" logico al porto di uscita collegato con la bobina A_+ della elettrovalvola A e che venga dato inizio all'estensione del pistone A. Il programma quindi giunge fino al passo successivo attraverso il cammino X_1 e Y_1.

Appena lo stantuffo comincia a muoversi gli stati dei sensori diventano $A_1=0$, $A_2=0$, $B_1=1$, $B_2=0$, e vengono "letti" in questo modo dalla micro unità di elaborazione.

Sottraiamo quindi il numero binario 0110 (la seconda condizione di riferimento) da questo numero e facciamo la "verifica dello zero" nel simbolo decisionale D3. Dato che la risposta non è zero (e che rimarà così per tutto il tempo in cui il pistone A è in movimento) lasciamo il simbolo decisionale D3 dalla uscita NO. Ancora una volta il calcolatore
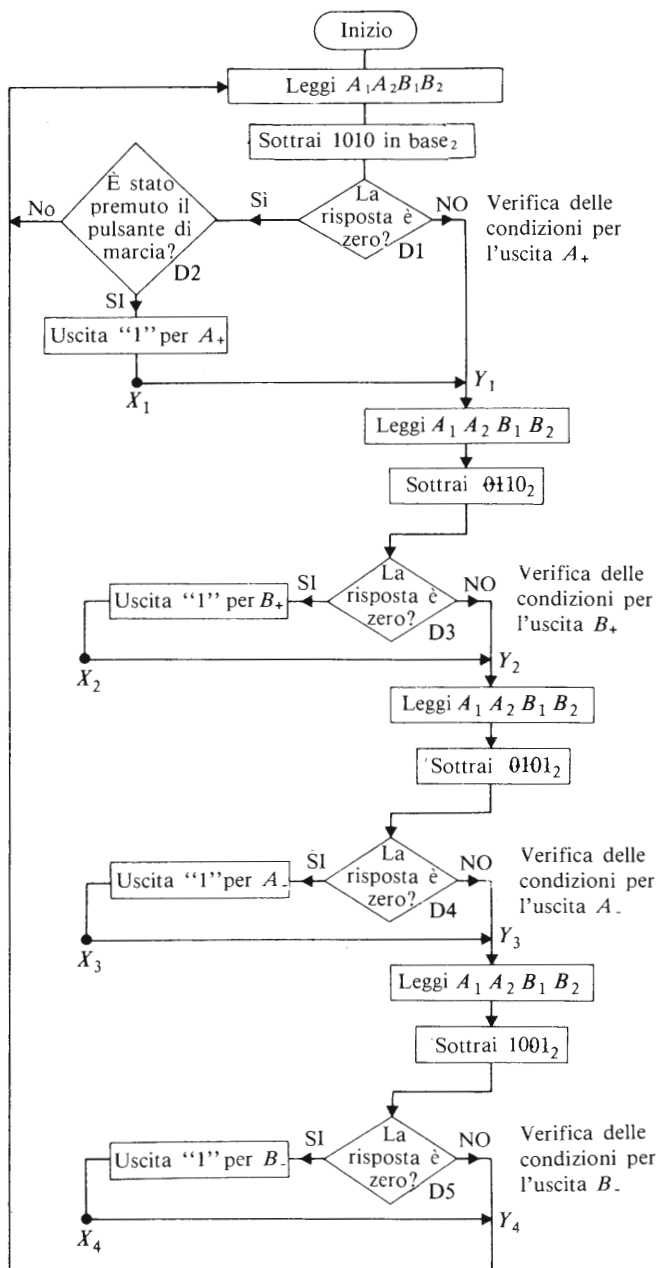
Fig. 11.7 Diagramma di flusso per il controllo della sequenza dei pistoni

legge gli stati di $A_1A_2B_1B_2$, ma questa volta in D4 paragona il valore col numero binario 0101; dato che il confronto dà risultato non nullo il programma procede attraverso l'uscita NO di D4 fino allo stadio successivo. Ancora una volta in D5 la risposta è NO, e si ritorna all'inizio del diagramma di flusso.

Il programma ora fa in modo che i passi riportati nel diagramma di flusso vengano ripetuti ciclicamente fino a che il pistone A non raggiunga la fine del suo movimento, quando gli stati dei sensori $A_1A_2B_1B_2$ sono 0110. A questo punto verifichiamo la presenza della condizione B_2 nell'elemento D3, la risposta è SI, cosicché noi mettiamo un segnale logico "1" in uscita dal porto uscente B_+; questo fa sì che il pistone B venga estratto. Il programma ritorna nel ciclo principale attraverso X_2 Y_2.

Mentre il pistone B è in movimento il programma fa sì che il diagramma di flusso venga continuamente percorso in modo ciclico. Quando il pistone B è completamente estratto ($A_1A_2B_1B_2=0101$), lasciamo il simbolo decisionale D4 dalla sua uscita SI e provochiamo l'inizio del movimento A_-. Applicando un ragionamento simile al resto del diagramma di flusso, vediamo che il processo qui descritto viene ripetuto fino a che il ciclo completo non sia stato eseguito. Dopo di che, il programma ripete la verifica per la condizione A_+ come è stato delineato in precedenza.

Il diagramma di flusso della Fig. 11.7 rappresenta solo una delle possibili soluzioni del problema, e invitiamo perciò il lettore a trovare soluzioni alternative.

Un vantaggio dello schema controllato dal software rispetto alla versione realizzata in hardware è la possibilità di includere test addizionali nel programma per controllare il funzionamento del sistema. Per esempio è possibile controllare che il sensore A_1 stia funzionando in modo corretto nella maniera seguente. Poco dopo aver applicato un segnale logico "1" al porto di uscita A_+ (vedi l'uscita SI dal simbolo decisionale D2), il pistone A comincia ad estendersi; durante l'intervallo di tempo in cui il pistone A è in "viaggio" lo stato dei sensori $A_1A_2B_1B_2$ dovrebbero essere 0010. Possiamo inserire una sezione addizionale nel diagramma di flusso tra i punti X_1 e Y_1 (vedi Fig. 11.7) per controllare che l'uscita dal sensore A_1 sia di fatto cambiata da "1" a "0". Se questo cambiamento non si verifica, possiamo assumere che il sensore A_1 è difettoso.

La modifica del diagramma di flusso riportato nella Fig. 11.8 richiede una telescrivente o una unità di visualizzazione video per visualizzare l'informazione: "il sensore A_1 è difettoso", e provocare la fermata del sistema.

Una stampa di questo tipo produrrebbe una riduzione del tempo ne-
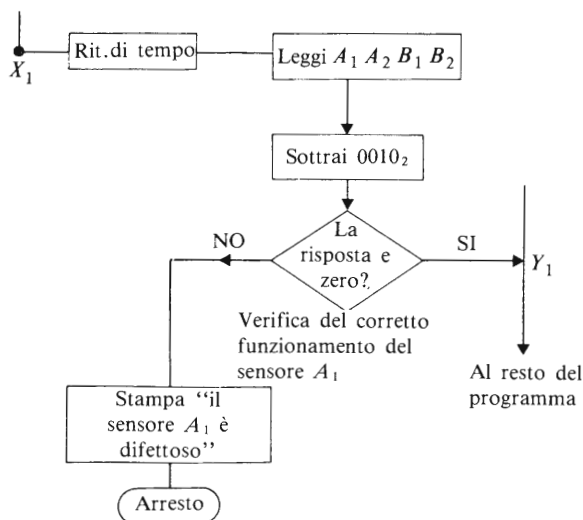
Fig. 11.8 Modifica del diagramma di flusso principale

cessario a rilevare il punto dove è avvenuto il difetto. Il ritardo *t* dovrebbe essere sufficientemente lungo per permettere al pistone di allontanarsi dalla posizione di ritrazione completa; il ritardo di tempo potrebbe essere introdotto per mezzo di un contatore nella microunità di elaborazione che conta uno specifico numero di impulsi di clock prima di permettere al sistema di leggere gli stati dei sensori. Simili modifiche per controllare il funzionamento degli altri sensori possono essere introdotti tra X_2 e Y_2; X_3 e Y_3; X_4 e Y_4.

Capitolo 12

Attrezzature di collaudo

12.1 Ricerca dei guasti nei circuiti digitali

Nei primi tempi dell'utilizzazione dei circuiti logici era frequente che il personale di manutenzione dovesse impiegare molte ore per localizzare un difetto anche relativamente semplice. Il tempo impiegato nella manutenzione dei circuiti elettronici è stato drasticamente ridotto per mezzo di dispositivi del tipo descritto in questo capitolo. Lo sviluppo di tecniche digitali, compresi i microprocessori, ha dato origine alla produzione di un'ampia gamma di attrezzature specializzate per il collaudo dei circuiti digitali.

Quando un sistema digitale non funziona in modo corretto è necessario isolare la parte difettosa il più rapidamente possibile. Se il sistema è complesso il tecnico di assistenza dapprima isola la piastra difettosa, dopo di che il componente difettoso può essere localizzato (nel frattempo la piastra difettosa viene sostituita con una piastra funzionante). Focalizzeremo la nostra attenzione sull'attrezzatura fondamentale di collaudo e più avanti ci occuperemo delle apparecchiature più sofisticate.

Come punto di riferimento generale, si consiglia il lettore di non utilizzare test di continuità sui circuiti logici, dato che essi non forniscono risultati utili e possono perfino danneggiare alcuni circuiti.

12.2 Sonde e "pinze" logiche

I difetti comuni dei sistemi logici includono porte non funzionanti che danno una uscita continua di un "1" o di uno "0", o porte che hanno un difetto su una linea di ingresso (per esempio uno "0" continuo in ingresso di una porta NAND fa sì che l'uscita sia un "1" continuo), o ancora porte che non seguono la tabella della verità, ecc...

Un metodo per controllare i tipi di difetti appena descritti è quello di inserire un segnale logico noto nella porta in questione, e di osservare la risposta su un indicatore appropriato. Il metodo più semplice per inseri-
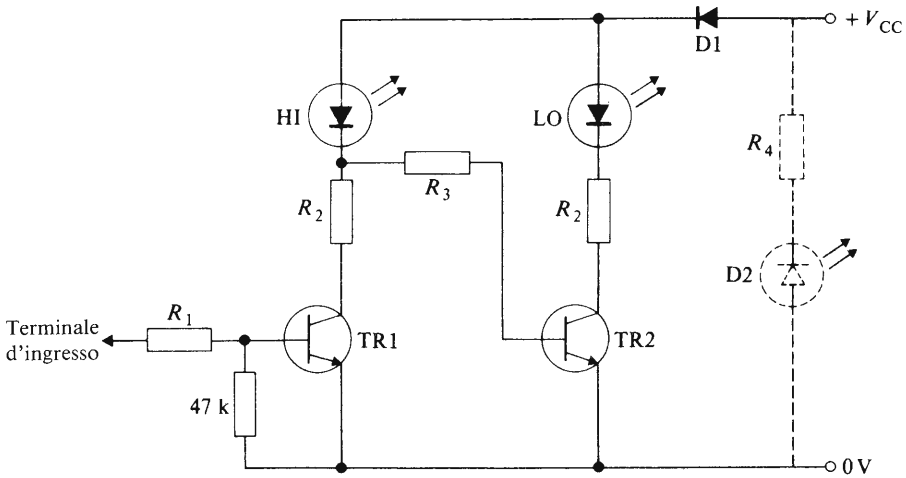
Fig. 12.1 Una semplice sonda logica

re un segnale di questo tipo si ottiene attraverso una sonda metallica connessa ad un livello logico appropriato (uno "0" o un "1"); l'uscita dalla porta è riportata su un monitor. In Fig. 12.1 è mostrata una semplice sonda realizzata a componenti discreti. La funzione del circuito è semplicemente quella di indicare la presenza o l'assenza di un segnale logico. Quando la sonda è applicata ad un segnale logico "1" (nella logica positiva), il transistore 1 è saturato e il diodo fotoemettitore HI (high = alto) è illuminato. Questo fa sì che la tensione del collettore del transistore 1 sia bassa, per cui il transistore 2 è interdetto e il diodo fotoemettitore LO (low = basso) è spento. Quando la sonda è applicata ad un potenziale basso (corrispondente ad uno "0" logico), il diodo fotoemettitore HI è spento, mentre viene illuminato il diodo fotoemettitore LO. La funzione del diodo D1 è quella di impedire alla sonda una connessione accidentale di una tensione inversa. Un'indicazione di tensione inversa può essere fornita collegando il diodo fotoemettitore D2 e la resistenza R_4 tra i terminali del generatore.

I valori dei componenti della Fig. 12.1 dipendono non soltanto dal valore della tensione del generatore V_{CC} ma anche dalle tensioni associate con i livelli logici che devono venire misurati. Valori tipici possono essere $R_2 + R_4 = 50 \times (V_{CC} - 2)\ \Omega$, $R_3 = 33\ \mathrm{k}\Omega$. In Fig. 12.2 si ha una soluzione alternativa che utilizza due porte NAND (metà di una integrata 7400).

I valori delle resistenze del circuito sono quelli tipici impiegati nei circuiti TTL.

Una sonda logica professionale è illustrata nella Fig. 12.3: essa è adatta per circuiti TTL, DTL o CMOS funzionanti con una alimentazione di
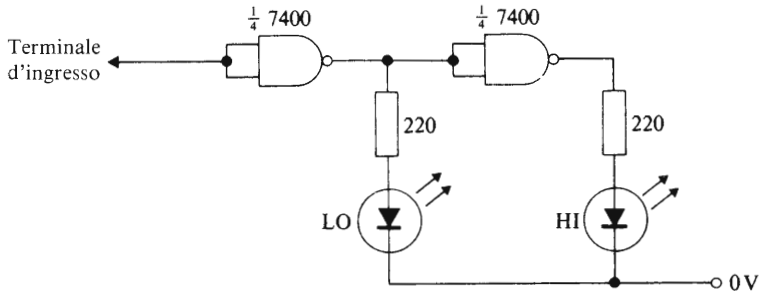
Fig. 12.2 Una sonda logica che utilizza porte NAND

5 V; un diodo fotoemettitore verde indica uno "0" logico e un diodo fotoemettitore rosso indica un "1" logico. Le sonde logiche hanno spesso un'ampia gamma di puntali (vedi figura); le sonde generalmente presentano un meccanismo di "allungamento d'impulso" che fa sì che un singolo impulso di brevissima durata (diciamo tra i 30 e i 50 nanosecondi) provochi l'illuminazione dei diodi fotoemettitori per un tempo abbastanza lungo da permettere all'occhio umano di vederlo. A seconda del tipo di sonda si possono fornire indicazioni su un circuito aperto o su un livello logico "cattivo" (cioè compreso tra "0" e "1") o su una sequenza di impulsi.

Come è stato precedentemente menzionato, un segnale di controllo può essere inserito nel sistema applicando un "1" logico o uno "0" logico per mezzo di una semplice sonda metallica. Sono disponibili attrezzature di collaudo sotto forma di generatori di impulso che possono inserire non solo livelli logici fissi come "0" o "1" ma singoli impulsi o una sequenza di impulsi. Questi dispositivi sono utili quando si eseguono verifiche sui contatori e sui registri a scorrimento. Nell'aspetto esteriore il generatore di impulsi è simile alla sonda di Fig. 12.3.

Un altro tipo di sonda è la cosiddetta "sonda di corrente" (current-tracer).

Quando è posta su un filo o su una pista di circuito stampato, essa indica per mezzo di un LED, se lungo la connessione stanno fluendo impulsi di corrente. Gli impulsi di corrente possono ad esempio essere facilmente iniettati per mezzo dell'impulsatore logico ricordato precedentemente.

Quando si teme che un circuito integrato sia guasto, diviene utile strumento di diagnosi la **pinza logica**. Tale dispositivo è fissato direttamente sull'integrato (vedi Fig. 12.4). La pinza mostrata è impiegabile per circuiti integrati a 14 o a 16 piedini e lo stato logico di ognuno di essi è in-

Fig. 12.3 Una sonda logica professionale (Riproduzione per gentile concessione della RS Components Ltd)

Fig. 12.4 Pinza logica (Logic clip o logic checker) (Riprodotto per concessione della Hewlett Packard Ltd)

dicato da un singolo diodo emettitore di luce. Ciò implica che l'utente può leggere contemporaneamente gli stati logici di tutti i piedini dell'integrato.

La pinza logica trae la sua alimentazione dai piedini di alimentazione dell'integrato sotto prova.

Se lo stato logico di uno dei piedini commuta a frequenza inferiore a circa 30 Hz il segnale appare sul LED corrispondente al piedino in questione con una luminosità percettibilmente pulsante.

Sopra ai 30 Hz il LED presenta una luce con brillanza media inferiore a quella associata con un segnale stabile "1".

Impiegata con un impulsatore logico, la pinza logica rappresenta un utile mezzo di verifica per il circuito stesso della tabella della verità di contatori e registri a scorrimento.

Quando il circuito integrato in prova viene individuato come guasto, esso può essere rimosso (dopo averlo dissaldato) mediante un attrezzo detto "estrattore": si tratta di un semplice utensile a molla con dentini che si collocano tra i piedini del circuito integrato e forniscono una sufficiente forza di aggancio per l'estrazione. Un utensile analogo all'estrattore è l'inseritore di circuiti integrati nel supporto stampato.

Qualora il circuito da realizzare impieghi un numero limitato di integrati può essere conveniente montare zoccoli sul circuito stampato.

In tal modo la sostituzione di un circuito integrato eventualmente guasto può essere compiuta facilmente, rapidamente, e senza in alcun modo danneggiare il supporto stampato come può avvenire nel caso di dissaldatura.

12.3 Analizzatori di stati logici

Per individuare una sezione guasta di una rete logica complicata, gli strumenti da impiegare devono essere più sofisticati di quanto siano quelli descritti precedentemente. L'**analizzatore di stati logici** è uno di questi. Esso possiede generalmente un visualizzatore realizzato mediante un tubo a raggi catodici (vedi Fig. 12.5).

L'analizzatore di stati logici è progettato per leggere lunghe sequenze di dati binari e rappresentarle sullo schermo del tubo a raggi catodici.

L'analizzatore opera fintanto che non appaia una particolare "parola" corrispondente ad una particolare situazione di errore (o di guasto), dopo di che esso presenta sul tubo a raggi catodici i dati successivamente giunti entro un predeterminabile intervallo di tempo. Tali dati vengono memorizzati dall'analizzatore permettendo così all'operatore di osservarli ed analizzarli con comodità.
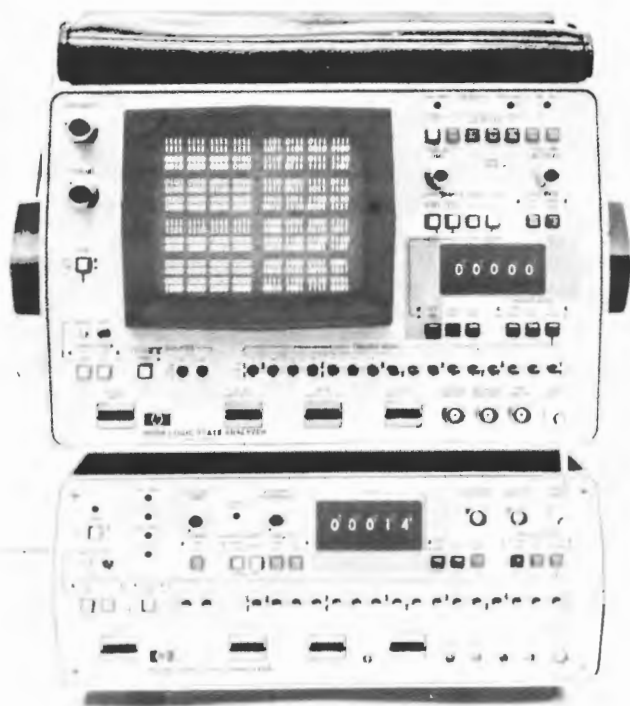
Fig. 12.5 Un analizzatore di stati logici (Riprodotto per concessione della H.P. Ltd)

12.4 Strumenti di misura analogici

Uno strumento di misura analogico possiede un indicatore a bobina mobile e pertanto può dare informazione in ogni punto della scala di misura. Un tester convenzionale a molte scale è un esempio di strumento analogico.

Gli strumenti analogici sono circuitalmente abbastanza semplici e molto affidabili; inoltre essi non richiedono normalmente un sistema di alimentazione per funzionare (tranne che nel caso in cui si debbano misurare resistenze per cui debbono essere impiegate batterie o altri sistemi di

alimentazione). Essi possono essere utilizzati per indicare (misurare) ampere, volt, ohm. Questi strumenti sono impiegati nei circuiti logici per misurare livelli di tensione e di corrente continua; una loro caratteristica è che in termini di circuiti elettronici la loro resistenza interna è piuttosto bassa: ciò può rappresentare un grande o piccolo inconveniente a seconda delle resistenze del rimanente del circuito. Per maggiori dettagli sui limiti di impiego di uno strumento che possieda una bassa resistenza interna il lettore è rinviato al testo *Elementi di elettronica teorica e pratica* (Hoepli).

Un'altra caratteristica di un voltmetro convenzionale a bobina mobile, che in molti casi rappresenta uno svantaggio in confronto con le prestazioni di uno strumento elettronico, è rappresentata dal fatto che, nella maggior parte dei casi, il fondo scala di tali strumenti è dell'ordine di un volt.

Se risulta necessario misurare valori di tensione bassi il voltmetro deve essere impiegato insieme ad un amplificatore. Se la resistenza interna o la bassa sensibilità di un voltmetro convenzionale rappresenta un problema, può essere preferibile impiegare un voltmetro elettronico. Esso consiste in uno strumento a bobina mobile preceduto da un opportuno amplificatore. La resistenza interna di un voltmetro elettronico è tipicamente di 10 MΩ o maggiore, e il campo di frequenze che possono essere misurate varia dalla continua ad alcuni megahertz.

12.5 Voltmetri digitali (D.V.M.)

Uno strumento digitale ha un indicatore di tipo digitale (decimale) a LED, a cristalli liquidi, a scarica nei gas, a filamento o altro ancora.

Va posta attenzione nella scelta dello strumento digitale dato che le specifiche possono trarre in inganno gli inesperti. Il visualizzatore è specificato in termini del massimo numero di cifre: ad esempio alcuni strumenti con quattro cifre sono capaci di fornire una lettura massima di 9999, mentre in altri la massima lettura è solo di 1999. Il primo tipo viene indicato come un *misuratore a 4 cifre* mentre il secondo come uno a $3\frac{1}{2}$. La precisione di molti strumenti digitali è sovente espressa come ± 1 o ± 2 sulla posizione meno significativa del visualizzatore. Nel caso di un sistema a tre cifre (massimo 999) un errore di ± 1 in una lettura di 999 rappresenta un errore dello 0,1% soltanto!

Tuttavia se l'utilizzatore intende misurare 10 volt nel campo di misura del 999 volt, l'errore $\pm$ 1 rappresenta un errore del $\pm$ 10%. E' spesso consigliabile impiegare uno strumento digitale in cui possano essere tra-

scurate una o due cifre nella parte meno significativa.

Vi sono numerosi schemi di strumenti digitali; ne descriveremo uno (Fig. 12.6).

Una tensione da misurare V_{in} viene applicata a un comparatore unitamente ad una seconda tensione V_2. Se il valore di V_{in} è maggiore di quello di V_2, l'uscita del comparatore è nello stato logico "1". Tale uscita apre la porta AND alla quale è inviato il segnale di clock.

Supponiamo che $V_{in} = 18$ V e inizialmente $V_2 = 0$.
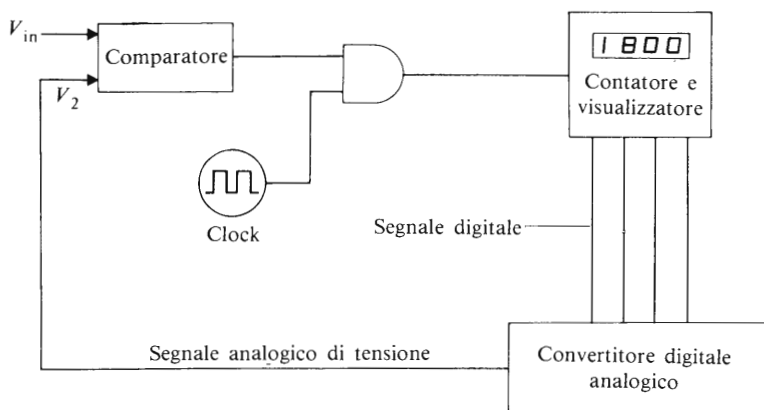
Fig. 12.6 Uno schema a blocchi di un voltmetro digitale

L'uscita del comparatore sarà quindi uguale a un "1" logico. Ciò permette agli impulsi di clock di venire applicati al contatore. L'uscita dal contatore viene rappresentata su un sistema di visualizzazione e contemporaneamente applicata a un convertitore digitale analogico (D.A.C. = digital to analogue converter): quest'ultimo trasforma il segnale digitale del contatore nel segnale di tensione analogico V_2.

Supponiamo che il D.A.C. fornisca un'uscita di 0,01 V per ogni cifra binaria di ingresso. Dopo 1800 impulsi di clock l'uscita del D.A.C. sarà pertanto 18 V, cioè $V_2 = V_{in}$.

Quando si raggiunge questa condizione l'uscita del comparatore commuta a zero impedendo che ulteriori impulsi di clock raggiungano il contatore.

Il numero visualizzato può essere calibrato in termini di tensione d'ingresso. Il tempo impiegato per la conversione della tensione d'ingresso in una espressione digitale dipende dalla frequenza del clock. Se esso è

ad esempio a 1 kHz con un ingresso di 18 V verranno impiegati 1,8 s (1800 impulsi per 0,001 s); a 100 kHz tale tempo si ridurrebbe a 0,018 s.

A seconda delle frequenze di clock, perciò, il voltmetro digitale sopra descritto può funzionare a diverse velocità.

In generale un dispositivo basato sul principio sopra descritto prende il nome di *convertitore analogico digitale* (A.D.C. = analogue-to-digital converter).

Il problema di una conversione analogico digitale in presenza di rumore a frequenza di rete può essere eliminato adottando circuiti che abbiano una elevata reiezione al rumore. Fra questi, uno dei più comuni è il circuito noto come convertitore analogico digitale a **doppia rampa**.

La maggioranza dei multimetri digitali a circuiti integrati funziona sul principio della doppia rampa (vedi anche il paragrafo 12.6).

12.6 Multimetri digitali

L'elemento base di un multimetro digitale è il voltmetro digitale. La corrente viene misurata impiegando un voltmetro digitale in unione con un'opportuna resistenza R detta di "shunt" posta in parallelo al voltmetro stesso. Se ai capi della resistenza di shunt R nella quale fluisce la corrente I si sviluppa la tensione V, vale la relazione:

$$V = \text{lettura del voltmetro} = IR = \text{corrente da determinare} \times R$$

Se si impiega una resistenza di shunt da 0,2 Ω, la lettura del voltmetro digitale è 0,2 × la corrente I del circuito (misurata in ampere). E' possibile perciò calibrare l'indicazione dello strumento in termini di corrente che fluisce nel circuito. Purché il voltmetro digitale sia stato opportunamente modificato per misurare tensioni alternate (vedi più avanti), questo sistema è applicabile alla misura di corrente alternata.

Il valore di una resistenza sconosciuta R può essere ottenuto misurando la tensione V ai capi dello strumento, quando in esso fluisce una corrente nota I.

$$V = \text{indicazione del voltmetro} = I \times \text{il valore incognito della resistenza}$$

Per esempio se $I = 1$ mA, un voltmetro digitale con fondo scala 2 V è calibrato per un campo di resistenze da 0 a 2 kΩ.

La calibrazione dello strumento è compiuta dal costruttore ed è auto-

maticamente mantenuta quando si cambia il campo di misura dello strumento.

La Fig. 12.7 illustra un circuito integrato a 40 piedini, in contenitore plastico, che incorpora tutta la circuiteria elettronica necessaria per pilotare uno strumento digitale da pannello. L'integrato mostrato è realizzato con tecnologia CMOS ed è proposto per la misura di tensioni continue.

Fig. 12.7 Un circuito integrato per uno strumento digitale da pannello (riprodotto per concessione della RS Components Ltd)

Esso richiede per la restaurazione di un voltmetro digitale un visualizzatore a 3 $\frac{1}{2}$ cifre, una alimentazione e alcuni altri componenti elementari. L'integrato della Fig. 12.7 impiega il principio della doppia rampa e fornisce inoltre un'indicazione automatica di polarità; una prestazione tipica di questo dispositivo e di molti altri integrati per voltmetri digitali è l'indicazione di superamento del fondo-scala (over-range) e l'auto azzerramento dello scorrimento (il visualizzatore indica zero se i terminali di ingresso sono cortocircuitati).

Per poter leggere tensioni elevate è necessario introdurre un opportuno circuito attenuatore tra la tensione di ingresso e l'integrato; un tipico attenuatore è mostrato nella Fig. 12.8.

Il coefficiente di attenuazione è dato per il circuito indicato dalla relazione

$$\text{Coefficiente di attenuazione} = \frac{R_2}{R_1 + R_2}.$$

Il resistore R_1 è normalmente fisso (è consigliabile un resistore ad ossido con il 2% di tolleranza) e R_2 è composta di un resistore di valore fisso R_3 (di caratteristiche analoghe a quelle di R_1) in serie con una resi-
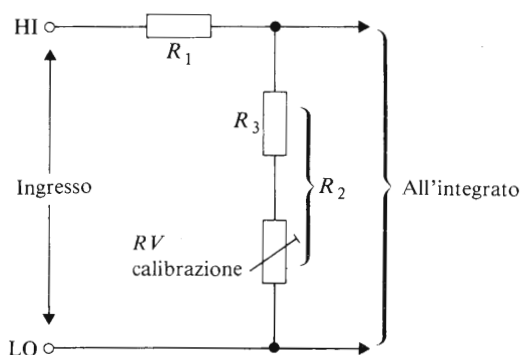
Fig. 12.8 Un attenuatore di tensione

stenza variabile necessaria per la calibrazione dell'attenuatore, (''Trimmer'' a 20 giri in cermet). Valori tipici per l'attenuazione 10:1 sono:

$$R_1 = 910 \text{ k}\Omega; \ R_3 = 91 \text{ k}\Omega; \ RV = 20 \text{ k}\Omega$$

Questi valori aumentano il valore di fondo scala di un voltmetro digitale da 2 V a 20 V.

La misura di tensioni alternate può essere compiuta mediante raddrizzamento della alternata prima della sua applicazione al voltmetro digitale; a tale scopo è bene usare un raddrizzatore di precisione realizzato con amplificatori operazionali. Va però notato che lo strumento è calibrato con una forma sinusoidale; se il segnale da misurare non è sinusoidale la tensione indicata non rappresenta l'effettivo valore (efficace).

12.7 Misura digitale della frequenza

Lo schema base di molti misuratori digitali di frequenza è indicato nella Fig. 12.9. Il segnale di cui si deve misurare la frequenza è applicato a un comparatore elettronico, che lo trasforma in un treno di impulsi logici alla stessa frequenza del segnale di ingresso.

Gli impulsi vengono a loro volta inviati all'ingresso di un contatore tramite una porta AND che ha come secondo ingresso l'uscita di uno oscillatore molto stabile (normalmente un oscillatore controllato al quarzo). Il contenuto del contatore è rappresentato su un opportuno visualizzatore. Con riferimento alla Fig. 12.9, l'impulso di durata T prodotto da un oscillatore costituisce la base temporale: durante questo intervallo
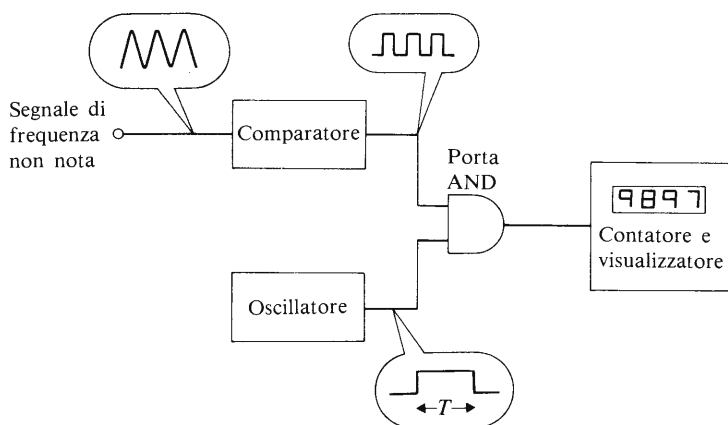
Fig. 12.9 Uno schema a blocchi di un misuratore di frequenza

di tempo viene contato il numero di impulsi del segnale a frequenza non nota.

Se il periodo T è di 1 secondo e ad esempio il segnale sotto test fornisce 9897 impulsi al secondo, il contatore visualizzerà tale numero alla fine del periodo di misura.

In apparecchi commerciali il periodo di conteggio viene continuamente ripetuto, cosicché, quando il segnale di ingresso cambia la propria frequenza, cambia anche il numero visualizzato che la rappresenta.

12.8 L'oscilloscopio a raggi catodici

Un oscilloscopio a raggi catodici è un elemento indispensabile per il tecnico di un reparto di manutenzione. Esso gli fornisce un "occhio" elettronico per "guardare" forme d'onda e livelli di tensione all'interno di un circuito elettronico.

Un oscilloscopio a raggi catodici contiene un tubo a raggi catodici e il suo controllo; si chiama **tubo a singolo fascio elettronico** (single beam) un tubo a raggi catodici capace di presentare una sola forma d'onda alla volta; la maggioranza degli oscilloscopi possiede tubi di questo tipo. Un tubo a raggi catodici che può presentare due forme d'onda contemporaneamente si chiama **tubo a doppio fascio elettronico** (twin beam); questo tipo di tubo è particolarmente utile quando devono essere studiate le relazioni esistenti fra due forme d'onda. Speciali tipi di tubi a raggi catodici possono fornire anche più di due tracce, talvolta di differenti colori.

Il tempo impiegato dalla traccia luminosa per sparire è noto come **persistenza** dello schermo. La maggior parte degli oscilloscopi per circuiti digitali impiega tubi a *breve persistenza.* Quando abbiamo necessità di osservare forme d'onda a frequenza molto bassa, come può avvenire ad esempio nel monitoraggio cardiaco, si impiegano tubi a *lunga persistenza*.

Ciò permette all'operatore di osservare la forma d'onda per svariati secondi prima che essa svanisca.

Una alternativa, sia nelle tecniche digitali sia in quelle analogiche, è l'impiego di **tubi a memoria** (storage tube) i quali possono "immagazzinare" l'immagine di una forma d'onda per un tempo pressoché indefinito.

Nella Fig. 12.10 è rappresentato il pannello frontale di un tipico oscilloscopio a raggi catodici. I comandi sono raggruppati in una opportuna parte; quelli di luminosità e di fuoco controllano rispettivamente la intensità della traccia e la sua definizione. Sullo schermo del tubo a raggi catodici viene posto un *reticolo* avente normalmente le maglie di 1 cm di lato. Unitamente ai comandi per l'asse X e per l'asse Y il reticolo permette l'impiego dell'oscilloscopio come strumento di misura.

I comandi nella parte sinistra del pannello frontale sono associati al segnale da presentare sullo schermo (segnale Y). Il comando *traslazione Y* permette all'utilizzatore di muovere la traccia in alto o in basso sullo schermo. Il comando *volt/cm* permette all'utilizzatore di modificare il guadagno dell'amplificatore Y in modo da rappresentare la forma d'onda più o meno amplia. Solitamente si inizia a lavorare con questo comando posto al valore massimo (minimo guadagno dell'amplificatore) e lo si riduce progressivamente fino a che si ottiene una rappresentazione soddisfacente. La funzione del commutatore D.C./A.C. è la seguente: nella posizione D.C. il segnale è trasmesso completo all'amplificatore (sia la componente continua sia quella alternata), mentre nella posizione A.C. soltanto la componente alternata del segnale viene visualizzata sullo schermo del tubo. Il segnale di ingresso viene collegato tra la coppia di terminali posta nella parte inferiore sinistra del pannello frontale. I comandi nella parte destra si riferiscono all'asse X (o base dei tempi) dell'oscilloscopio.

Il comando *traslazione X* permette all'utilizzatore di spostare la traccia a sinistra o a destra, mentre il "comando variabile della base dei tempi" e quello del "tempo/cm" permette all'utilizzatore di modificare la velocità con cui il punto luminoso si sposta automaticamente da sinistra a destra. Al terminale "segnale di sincronismo esterno" si può, se lo si desidera, collegare un segnale che provoca la partenza della traccia
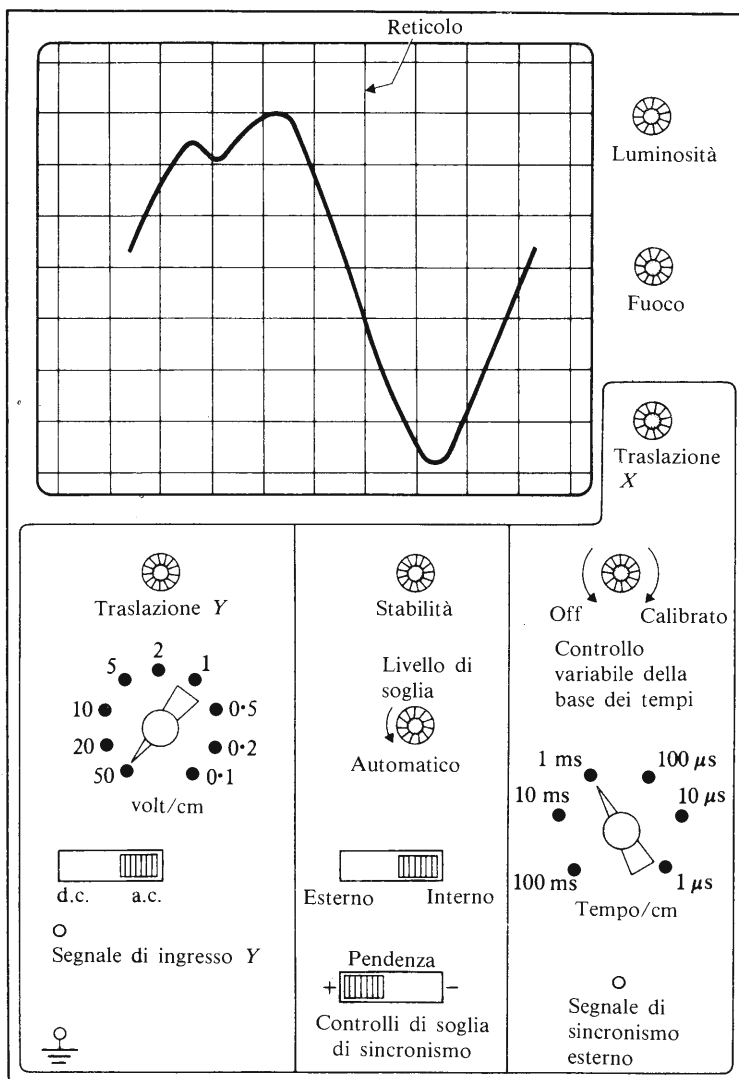
Fig. 12.10 Un tipico pannello frontale di un oscilloscopio

luminosa. Normalmente non è necessario applicare un segnale a questo terminale.

I comandi della parte centrale del pannello frontale permettono all'utilizzatore di scegliere il punto di soglia sulla forma d'onda da osservare. Il comando "livello di soglia" è normalmente posto su "automatico" e il comando "stabilità" viene regolato finché una forma d'onda stabile appare sullo schermo dell'oscilloscopio. Quando i comandi precedenti sono stati posizionati, l'avvio della traccia è ancora controllato dalla scelta della "pendenza" (posizione + o —). Nella posizione + (vedi Fig. 12.10) la forma d'onda viene rappresentata a partire da un punto in cui la sua pendenza sia positiva. L'interruttore interno/esterno è posto normalmente nella posizione interno. La posizione "esterno" indica che la forma d'onda sullo schermo dell'oscilloscopio deve essere sincronizzata con il segnale applicato all'ingresso "segnale di sincronismo esterno" (parte inferiore destra del pannello frontale).

Servizio radiotecnico:
Volume I: "Strumenti per radiotecnici". Strumenti per la messa a punto e la riparazione degli apparecchi radio. 16ª edizione aggiornata a cura di *Giorgio Terenzi*. In-16, di pagine XVI-368, con 212 figure nel testo e 4 tavole fuori testo. Copertina a colori plasticata e telata
ISBN 22-203-1088 L. 6000
Volume II: "Radio riparazioni". Ricerca ed eliminazione dei guasti e difetti negli apparecchi radio. 17ª edizione aggiornata a cura di *Giorgio Terenzi*. In-16, di pagine XII-444, con 240 figure nel testo. 6 tavole fuori testo e 11 tabelle. Copertina a colori plasticata e telata
ISBN 88-203-1127-5 L. 8000

L'apparecchio radio ricevente e trasmittente. Aspetti fondamentali - principio di funzionamento dell'apparecchio ricevente - La modulazione di frequenza - Ricevitori autocostruiti per onde corte - Ricevitori professionali per onde corte - Antenne - Trasmettitori per dilettanti -Trasmettitori professionali - Radiotelefoni. 6ª edizione aggiornata a cura di *G. Terenzi*. In-8, di pagine XXIV-320, con 247 figure nel testo e 9 tavole fuori testo. Copertina a colori plasticata **ISBN 88-203-1155-0 L. 14000**

L'apparecchio radio ricevente e trasmittente. Aspetti fondamentali - Principio di funzionamento dell'apparecchio ricevente - La modulazione di frequenza - Ricevitori autocostruiti per onde corte - Ricevitori professionali per onde corte - Antenne - Trasmettitori per dilettanti - Trasmettitori professionali - Radiotecnici. 5ª edizione interamente rifatta ed aggiornata a cura di *Giorgio Terenzi*. In-8, pagine XXIV-280 con 222 figure nel testo e 7 tavole fuori testo. Copertina a colori plasticata e telata
ISBN 88-203-0910-6 L. 10000

Primo avviamento alla conoscenza della radio. Come è fatto, come funziona, come si adopera l'apparecchio radio, come si possono costruire apparecchio radio a transistori e a valvole, a circuiti integrati. 21ª edizione ampiamente riveduta e aggiornata a cura di *Giorgio Terenzi*. In-16, di pagine XII-396, con 243 figure e 51 schemi di apparecchi radio di facile costruzione. Copertina a colori plasticate e telata
ISBN 88-203-1090-2 L. 6000

Il videolibro. Televisione pratica in bianco-nero ed a colori. 10ª edizione ampliata ed aggiornata a cura di *Giorgio Terenzi*. In-8, di pagine XXIV-706, con 590 figure, 36 tavole fuori testo con schemi di televisori in bianco e nero ed a colori. 19 tavole fuori testo a colori. Copertina a colori plasticata
ISBN 88-203-1132-1 L. 20000

Strumenti per videotecnici. L'oscilloscopio e gli altri strumenti per il servizio videotecnico. 5ª edizione aggiornata a cura di *Giorgio Terenzi*. In-8, di pagine XII-332, con 281 figure e XX tavole fuori testo. Copertina a colori plasticata e telata **ISBN 88-203-1116-X L. 10000**

L'audiolibro. Amplificatori - Altoparlanti - Microfoni - Giradischi - Registratori magnetici. 8ª edizione aggiornata ed ampliata a cura di *Giorgio Terenzi*. In-8, di pagine XXIV-372, con 314 figure di cui 62 schemi elettrici, 15 tavole fuori testo. Copertina a colori plasticata e telata
ISBN 88-203-1072-4 L. 12000

Radio elementi. Corso preparatorio per radiotecnici e riparatori - Elementi generali di elettricità - Elementi generali di radiotecnica - Parti componenti l'apparecchio radio ricevente - Teoria e pratica delle valvole radio e dei transistori - Schemi e dati costruttivi di apparecchi radio a cristallo e di piccoli apparecchi a valvole per dilettanti - Apparecchi a transistor a FET, a circuiti integrati - Alimentatori e altoparlanti - Schemi e dati pratici per la costruzione di apparecchi radio riceventi ad uso dei dilettanti - Ricevitori AM/FM e FM stereo - Norme per la taratura delle supereterodine. 11ª edizione riveduta ed aggiornata a cura di *Giorgio Terenzi*. In-16 di pagine XXXII-398, con 262 figure e 14 tavole fuori testo. Copertina a colori plasticata e telata **ISBN 88-203-1079-1 L. 7000**

Servizio videotecnico. Verifica, messa a punto e riparazione dei televisori a valvole, a transistor, a circuiti integrati, in bianco e nero e a colori. 8ª edizione riveduta e aggiornata a cura di *Giorgio Terenzi*. In-8, di pagine XXIV-476, con 394 figure di cui 33 a colori e 32 tavole. Copertina a colori plasticata **ISBN 88-203-1186-0 L. 20000**

EDITORE ULRICO HOEPLI MILANO

I progressi dell'elettronica hanno condotto allo sviluppo esplosivo dell'elettronica digitale. I sistemi digitali si sono evoluti a un tal punto che l'industria richiede oggi al computer funzioni fino a pochi anni fa svolte dai dispositivi elettromeccanici. Con questo testo l'Autore viene incontro alla necessità di informazione nel campo dell'elettronica digitale, raggiungendo un giusto equilibrio tra le funzioni, complementari ed entrambi essenziali per studenti e per tecnici, di apprendimento della teoria e di avviamento all'applicazione pratica.

Sono trattati i seguenti argomenti: dispositivi semiconduttori e circuiti integrati (Capitoli 1-10); porte logiche (Capitoli 2-4); elementi bistabili e sistemi di memoria (Capitolo 5 e 9); numeri binari, contatori e registri di scorrimento (Capitoli 6-8); calcolatori e microprocessori (Capitolo 11); attrezzature di collaudo (Capitolo 12).

Il testo trova un suo pubblico preciso negli studenti degli Istituti tecnici statali e delle Scuole professionali per l'industria e l'artigianato, nelle quali sia svolto il corso di elettronica digitale.

L. 7.000

ISBN 88-203-1243-3